

Grado Universitario en Ingeniería Informática
2017-2018

Trabajo Fin de Grado

“RECONOCIMIENTO DEL ESTADO DE ÁNIMO DE UN CONDUCTOR UTILIZANDO TÉCNICAS DE VISIÓN POR COMPUTADOR”

Juan Luis Cazorla Aláez

Tutor

Arturo de la Escalera Hueso

Leganés, 2018

Reconocimiento del estado de ánimo de un conductor utilizando técnicas de visión por computador

por

Juan Luis Cazorla Aláez

Resumen

Con este proyecto, se ha desarrollado un Sistema Avanzado de Asistencia a la Conducción (del inglés, ADAS) capaz de monitorizar el rostro de un conductor y determinar su estado de ánimo a través de las imágenes. El fin del sistema es prever situaciones de estrés y de distracción al volante para tratar de reducirlas. Con la información suficiente sobre el conductor de un vehículo, se podría llegar a tomar acciones dentro del propio vehículo para adaptar la conducción. Por otro lado, si además se posee información que pueda determinar si el conductor se encuentra distraído o en una situación de estrés, podría adaptarse la conducción a esta situación para minimizar esto y mejorar la situación anímica del conductor en ese momento. De este modo, la cantidad de accidentes de tráfico podrían verse reducidos; ya que es el factor humano, principalmente por las distracciones al volante, uno de los factores más importantes en la causa de estos. Para cumplir con su funcionalidad, el sistema usa una combinación de tecnologías: por un lado, visión por computador para extraer información de las imágenes que se toman; por el otro, aprendizaje automático para determinar el estado de ánimo de la imagen obtenida a partir de la información anteriormente extraída.

Palabras clave: Visión por computador, Aprendizaje automático, Reconocimiento facial, Sistemas avanzados de asistencia a la conducción.

Índice

1.	Introducción	1
1.1	Motivación	1
1.2	Objetivos	4
1.3	Estructura de la memoria.....	5
2.	Estado del arte	6
2.1	Sistemas avanzados de asistencia a la conducción.....	6
2.1.1	Tecnologías de Visión Artificial para ADAS.....	7
2.1.2	Principales Funciones ADAS	9
2.2	Reconocimiento del estado de ánimo.....	12
2.2.1	Bismart Face and Emotion Recognition	12
2.2.2	Afectiva	13
3.	Análisis del sistema	15
3.1	Requisitos.....	15
3.1.1	Requisitos de usuario	17
3.1.2	Requisitos Software.....	22
3.2	Casos de uso.....	31
3.3	Marco regulador.....	35
4.	Arquitectura y Diseño	36
4.1	Arquitectura	36
4.1.1	Módulo de entrenamiento.....	37
4.1.2	Módulo Cliente.....	37
4.1.3	Modulo Servidor.....	37
4.2	Estudio de la solución final.....	38
5.	Implementación y despliegue	40
5.1	Implementación	40
5.1.1	Generar modelo predictivo	42
5.1.2	Funcionamiento del sistema	48
5.1.3	Cliente: manejo de la cámara.....	50
5.1.4	Servidor	52

5.1.5	Comunicación entre Cliente y Servidor	54
5.2	Despliegue	55
6.	Verificación, validación y evaluación	57
6.1	Pruebas de verificación	57
6.2	Pruebas de validación	68
7.	Planificación y presupuesto	73
7.1	Planificación	73
7.1.1	Justificación de la metodología	73
7.1.2	Tiempo estimado.....	75
7.2	Presupuesto	77
7.2.1	Costes parciales.....	77
7.2.2	Costes totales	78
7.3	Impacto socioeconómico	80
8.	Conclusiones y trabajos futuros.....	82
8.1	Conclusiones.....	82
8.2	Trabajos futuros	84
9.	Bibliografía	85
	Abstract	89
	Resumen.....	102

Índice de tablas

Tabla 1. Tabla de especificación de Requisitos.	15
Tabla 2. Requisito de Capacidad RUC-01	17
Tabla 3. Requisito de Capacidad RUC-02	17
Tabla 4. Requisito de Capacidad RUC-03	18
Tabla 5. Requisito de Capacidad RUC-04	18
Tabla 6. Requisito de Capacidad RUC-05	18
Tabla 7. Requisito de Capacidad RUC-06	18
Tabla 8. Requisito de Capacidad RUC-07	19
Tabla 9. Requisito de Capacidad RUC-08	19
Tabla 10. Requisito de Capacidad RUC-09	19
Tabla 11. Requisito de Restricción RUR-01	20
Tabla 12. Requisito de Restricción RUR-02	20
Tabla 13. Requisito de Restricción RUR-03	20
Tabla 14. Requisito de Restricción RUR-04	21
Tabla 15. Requisito de Restricción RUR-05	21
Tabla 16. Requisito de Restricción RUR-06	21
Tabla 17. Requisito Funcional RSF-01	22
Tabla 18. Requisito Funcional RSF-02	22
Tabla 19. Requisito Funcional RSF-03	23
Tabla 20. Requisito Funcional RSF-04	23
Tabla 21. Requisito Funcional RSF-05	23
Tabla 22. Requisito Funcional RSF-06	23
Tabla 23. Requisito Funcional RSF-07	24
Tabla 24. Requisito Funcional RSF-08	24
Tabla 34. Requisito Funcional RSF-09	24
Tabla 25. Requisito Funcional RSF-10	24
Tabla 26. Requisito Funcional RSF-11	25
Tabla 27. Requisito Funcional RSF-12	25
Tabla 28. Requisito Funcional RSF-13	25
Tabla 29. Requisito Funcional RSF-14	25
Tabla 30. Requisito Funcional RSF-15	26
Tabla 31. Requisito Funcional RSF-16	26

Tabla 32. Requisito Funcional RSF-17	26
Tabla 33. Requisito Funcional RSF-18	26
Tabla 35. Requisito Funcional RSF-19	27
Tabla 36. Requisito no Funcional RSNF-01.....	28
Tabla 37. Requisito no Funcional RSNF-02.....	28
Tabla 38. Requisito no Funcional RSNF-03.....	28
Tabla 39. Requisito no Funcional RSNF-04.....	29
Tabla 40. Requisito no Funcional RSNF-05.....	29
Tabla 41. Requisito no Funcional RSNF-06.....	29
Tabla 42. Requisito no Funcional RSNF-07.....	29
Tabla 43. Requisito no Funcional RSNF-08.....	30
Tabla 44. Requisito no Funcional RSNF-09.....	30
Tabla 45. Requisito no Funcional RSNF-10.....	30
Tabla 46. Casos de uso.	31
Tabla 47. Caso de Uso CU-01	32
Tabla 48. Caso de Uso CU-02	32
Tabla 49. Caso de Uso CU-03	33
Tabla 50. Caso de Uso CU-04	33
Tabla 51. Caso de Uso CU-05	34
Tabla 52. Tests de algoritmos.....	46
Tabla 53. Ejemplo de tabla de prueba de verificación (VET-XX).	57
Tabla 54. Prueba de Verificación VET-01	58
Tabla 55. Prueba de Verificación VET-02	59
Tabla 56. Prueba de Verificación VET-03	59
Tabla 57. Prueba de Verificación VET-04	60
Tabla 58. Prueba de Verificación VET-05	60
Tabla 59. Prueba de Verificación VET-06	61
Tabla 60. Prueba de Verificación VET-07	61
Tabla 61. Prueba de Verificación VET-08	62
Tabla 62. Prueba de Verificación VET-09	62
Tabla 63. Prueba de Verificación VET-10	63
Tabla 64. Prueba de Verificación VET-11	63
Tabla 65. Prueba de Verificación VET-12	64
Tabla 66. Prueba de Verificación VET-13	64
Tabla 67. Prueba de Verificación VET-14	65

Tabla 68. Prueba de Verificación VET-15	65
Tabla 69. Prueba de Verificación VET-16	66
Tabla 70. Matriz de trazabilidad de pruebas VET y Requisitos Software.	67
Tabla 71. Ejemplo de tabla de prueba de validación (VAT-XX).	68
Tabla 72. Prueba de Validación VAT-01	69
Tabla 73. Prueba de Validación VAT-02	70
Tabla 74. Prueba de Validación VAT-03	70
Tabla 75. Prueba de Validación VAT-04	71
Tabla 76. Prueba de Validación VAT-05	71
Tabla 77. Prueba de Validación VAT-06	72
Tabla 78. Matriz de trazabilidad de pruebas VAT y Requisitos de Usuario.	72
Tabla 79. Tabla de estimación de tiempos.....	75
Tabla 80. Tabla de costes materiales.	77
Tabla 81. Tabla de costes de personal.	78
Tabla 82. Tabla de Coste Total del proyecto.	79

Índice de ilustraciones

Ilustración 1. Fallecidos por accidente de tráfico en función de la región (Fuente: www.revista.dgt.es).	1
Ilustración 2. Víctimas de accidentes de tráfico ciclistas, motociclistas y peatones (Fuente: www.revista.dgt.es).	2
Ilustración 3. Tologías de visión artificial ADAS: radar. (Fuente: www.ti.com).....	7
Ilustración 4. Tologías de visión artificial ADAS: Laser. (Fuente: www.azosensors.com)	8
Ilustración 5. Tologías de visión artificial ADAS: Laser. (Fuente: www.azosensors.com)	8
Ilustración 6. Tologías de visión artificial ADAS: cámara. Fuente: (http://www.ti.com/ltds/ti/applications/automotive/adas/overview.page)	9
Ilustración 7. ADAS de Cambio involuntario de Carril. (Fuente: www.autopista.es).....	10
Ilustración 8. ADAS de previsión de colisión frontal. (Fuente: www.autopista.es)	11
Ilustración 9. ADAS de previsión de colisión frontal. (Fuente: www.autopista.es)	11
Ilustración 10. Bismart Magic Mirror. (Fuente: www.bismart.com).....	13
Ilustración 11. Afectiva Automotive AI. (Fuente: www.affective.com)	14
Ilustración 12. Diagrama de arquitectura del sistema.	36
Ilustración 13. Imágenes de muestra de las categorías empleadas de la base de datos. De izquierda a derecha: enfado, disgusto y susto en la primera fila; alegría, tristeza y sorpresa en la segunda.	42
Ilustración 14. Imágenes de muestra de las categorías empleadas de la base de datos. De izquierda a derecha: neutral y desprecio.....	42
Ilustración 15. Imagen de muestra para detectar un rostro.....	43
Ilustración 16. Imagen de muestra con rostro detectado.	43
Ilustración 17. Ejemplo de Facial Landmarks	44
Ilustración 18. Matrices de confusión normalizadas para caso 1 y caso 2 para el algoritmo SVM de kernel lineal. Dónde: anger2 es la unión de enfado y disgusto, fear es miedo, happiness alegría, neutral es neutral, sadness tristeza, surprise sorpresa, anger enfado y disgust disgusto.	45
Ilustración 19. Imágenes de muestra: Enfadado y Disgustado.	45
Ilustración 20. Diagrama de ejecución del sistema.....	49
Ilustración 21. Resultado de ejecución.	51
Ilustración 22. Ejemplo de conexión cliente servidor.	54
Ilustración 23. Diagrama de desarrollo en cascada.	74
Ilustración 24. Diagrama de Gantt de la planificación del proyecto.....	76

1. Introducción

En este apartado se describe brevemente la motivación que llevó a la realización de este proyecto, así como los principales objetivos que se pretenden lograr con él. Por último, se explica la estructura del documento completo.

1.1 Motivación

En la actualidad, los accidentes de tráfico siguen siendo una de las principales causas de mortalidad. En el mundo fallecen alrededor de 1.250.000 personas a causa de estos accidentes, pero cabe destacar que el índice de mortalidad por estos accidentes varía en gran medida en función de la región geográfica. [1]



Ilustración 1. Fallecidos por accidente de tráfico en función de la región (Fuente: www.revista.dgt.es).

Como se puede observar en la *Ilustración 1*, la probabilidad de que un accidente de tráfico sea mortal es mucho mayor en regiones como África o el Mediterráneo Oriental que en Europa, por ejemplo. Esta diferencia se debe, en gran medida, a la antigüedad media de los automóviles de cada región, a los sistemas de seguridad que incluyen estos automóviles y a la falta de regulación de muchos países. Por ejemplo, algo tan básico para nosotros como es el uso del cinturón de seguridad no se hizo obligatorio en España hasta 1990, por lo que muchos coches de los años 80 ni siquiera incluían este elemento básico de seguridad pasiva. [2]

Hoy día sigue habiendo países en los que el uso de este elemento de seguridad aún no es obligatorio para todos los ocupantes de un vehículo, y en otros muchos países se usan diariamente turismos que carecen de este elemento de seguridad. [1]

Además, cabe mencionar que el 22% de las personas que fallecen por accidentes de tráfico son peatones que ni si quiera ocupaban el asiento de un vehículo.



Ilustración 2. Víctimas de accidentes de tráfico ciclistas, motociclistas y peatones (Fuente: www.revista.dgt.es).

Realizando cambios (que hoy día son básicos para los países occidentales) en todos estos países subdesarrollados o en vías de desarrollo, las cifras de mortalidad mundial tendrían una tendencia a mejorar muy clara.

Por otro lado, se deben seguir investigando y desarrollando nuevos sistemas que aporten seguridad a los vehículos de fabricación actual. Haciendo así que el uso y la incorporación de estos nuevos sistemas en los vehículos de fabricación actual sea generalizado.

Cabe destacar que el factor humano sigue siendo el origen del 90 % de los accidentes de tráfico, ya sea por distracción, conducción bajo los efectos de alcohol o drogas, exceso de velocidad, etc. Esto sugiere que gran parte de estos accidentes podrían evitarse con sistemas que ayudasen al conductor a reducir estas situaciones de riesgo potencial. [3]

1.2 Objetivos

El objetivo principal de este proyecto consiste en desarrollar un Sistema Avanzado para la Asistencia a la Conducción, que ayude a reducir las situaciones de distracción al volante del conductor de un turismo.

Este sistema monitorizaría a través de una cámara la cara del conductor, pudiendo determinar el estado de ánimo de éste. De esta forma, el vehículo tendría información sobre el conductor y podría determinar la predisposición de distracción de este; pudiendo llegar a tomar acciones en el interior del vehículo para reducir esta mala predisposición.

Por otro lado, destacar que el sistema, además de permitir generar una conducción más serena y con menos distracciones, pueda dar lugar a una mejora en la experiencia de la conducción del vehículo, ya que puede dar lugar a un alto grado de personalización y adaptación para cada usuario.

1.3 Estructura de la memoria

El documento sigue la siguiente estructura:

1. Introducción, en la que se explican la motivación y los objetivos del proyecto llevado a cabo, así como los puntos de los que consta.
2. Estado del arte, que incluye una descripción de los ADAS (sistemas de asistencia avanzada en la conducción) más comunes actualmente y de algunas de las herramientas de reconocimiento de ánimo con mayor presencia.
3. Análisis, donde se lleva a cabo una breve descripción del proyecto, se constituyen los requisitos y se presenta el marco regulador del proyecto.
4. Diseño, donde se detallan la solución tomada y el diseño completo del sistema.
5. Implementación y despliegue, donde se explican tanto la implementación de las principales partes del software, como los requisitos necesarios para su despliegue en un dispositivo.
6. Verificación, validación y evaluación, detalla distintos tests llevados a cabo para determinar si el software desarrollado cumple con los requisitos.
7. Planificación y presupuesto, que detalla la metodología de desarrollo empleada para llevar a cabo el proyecto, indicando los tiempos llevados a cabo para cada tarea; así como se exponen los costes del proyecto. Por último, en este apartado, se hace un análisis del impacto socioeconómico del proyecto.
8. Conclusiones y trabajos futuros, donde se explican las distintas conclusiones tomadas durante la realización del proyecto y se exponen trabajos futuros relacionados.

2. Estado del arte

En esta sección se lleva a cabo un estudio del estado del arte, donde se investigan y se analizan tecnologías similares. De esta forma, se ha procedido a investigar sobre los dispositivos de asistencia a la conducción; así como las tecnologías de reconocimiento del estado de ánimo.

2.1 Sistemas avanzados de asistencia a la conducción

Los Sistemas Avanzados de Asistencia en la Conducción, o más conocidos como ADAS por sus siglas en inglés (*‘Advanced Driver Assistance Systems’*), son mecanismos que ayudan al conductor durante la conducción de un automóvil. Su principal función suele ser la de incrementar la seguridad, tanto para los ocupantes del vehículo, como para el resto de usuarios de las vías. [4 y 5]

‘Las tecnologías de asistencia a la conducción, mediante visión artificial, permiten al vehículo ver y comprender el entorno en el que circula. Los sistemas ven, interpretan y analizan el entorno para detectar el potencial riesgo de la situación en la que se encuentra el vehículo, identificando tanto la vía, sus límites, como los usuarios de la misma, ya sean vehículos, peatones o ciclistas.’ [6]

Los sistemas ADAS requieren de una serie de sensores que permiten la visión artificial del vehículo. Estos sensores pueden trabajar de forma independiente o en combinación para mejorar su funcionalidad.

A continuación, se hará una breve descripción de las principales tecnologías usadas para la visión artificial en los ADAS.

2.1.1 Tecnologías de Visión Artificial para ADAS

Las principales tecnologías de visión artificial usadas para son el Radar, el Lidar y Cámaras.

- Radar: *“El sistema se fundamenta en ondas electromagnéticas para medir distancias, altitudes, direcciones y velocidades de objetos estáticos o móviles. El uso de ondas electromagnética con diversas longitudes de onda permite detectar objetos más allá del rango de otro tipo de emisiones (luz visible, sonido, etc.) [6].”*

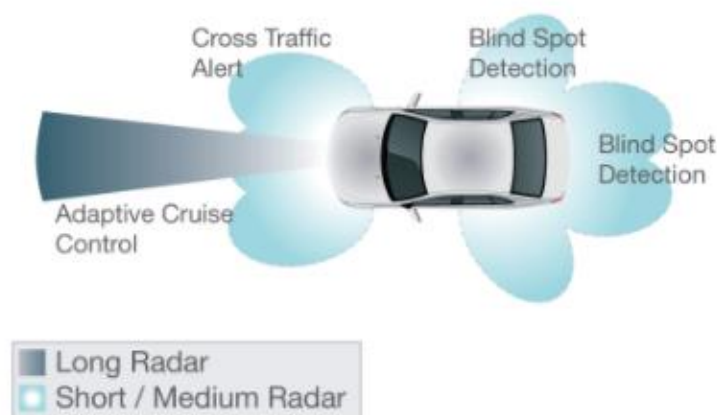


Ilustración 3. Tocologías de visión artificial ADAS: radar. (Fuente: www.ti.com)

Permite detectar la distancia, el ángulo y la velocidad de otros vehículos, así como a peatones y bicicletas.

- LIDAR o laser: *“Sus siglas provienen del acrónimo inglés Laser Imaging Detection and Ranging. La tecnología permite determinar la distancia desde un emisor láser a un objeto o superficie utilizando un haz láser pulsado. La medición se determina calculando el tiempo de retraso entre la emisión del pulso y su detección a través de la señal reflejada [6].”*

Permite detectar marcas viales, la posición de objetos reflectantes (como señales y matrículas), objetos móviles, los límites de la vía y si el espacio está libre para la circulación.

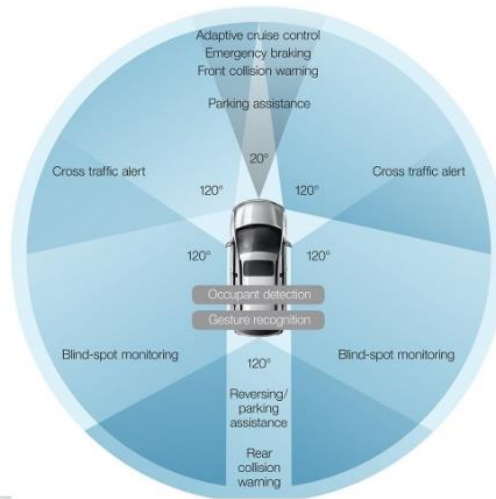


Ilustración 4. Tocologías de visión artificial ADAS: Laser. (Fuente: www.azosensors.com)

El LIDAR proporciona gran cantidad de información de todo el entorno, con resolución suficiente para identificar y categorizar objetos. De hecho, sigue funcionando correctamente en situaciones climáticas adversas.



LiDar sensing, like the Leddar technology, can detect multiple moving and static obstacles simultaneously and performs exceptionally well in adverse weather conditions.

Ilustración 5. Tocologías de visión artificial ADAS: Laser. (Fuente: www.azosensors.com)

- Cámaras: “Sistema basado en la captación y el reconocimiento del entorno mediante el procesamiento de la imagen. El ocular de la cámara percibe el entorno al igual que el ojo humano, percibiendo colores, texturas y formas y reconociendo los patrones de los principales elementos de la vía. Las cámaras pueden ser estereoscópicas o de un solo

objetivo, siendo su principal diferencia la forma de determinar la profundidad y distancias con los objetos del entorno [6].”

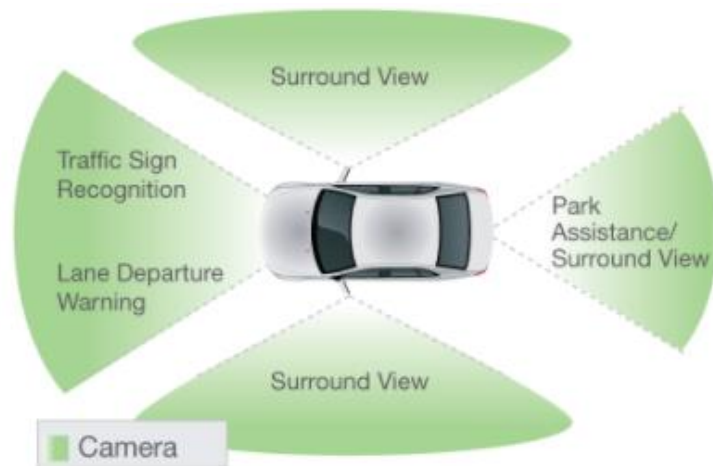


Ilustración 6. Tocologías de visión artificial ADAS: cámara. Fuente:
(<http://www.ti.com/lscs/ti/applications/automotive/adas/overview.page>)

Se puede usar para detectar a los demás vehículos, peatones, ciclistas, elementos de la vía e incluso a los propios usuarios del vehículo.

Actualmente es la tecnología con la que más aplicaciones se están investigando y desarrollando. Esto se puede deber por su coste reducido comparado con las anteriores, su alta durabilidad, la facilidad de integración e instalación y por la gran cantidad de información útil que puede obtener.

2.1.2 Principales Funciones ADAS

- Salida involuntaria de carril: hay principalmente dos funciones con impacto positivo en accidentes de este tipo. En el primer nivel el conductor recibe una alerta y mantiene el control del vehículo. En el segundo nivel el vehículo toma el control de la dirección del vehículo y lo adecua a la curvatura de la vía. Este sistema tiene impacto positivo también en colisiones laterales.

- Alerta de salida de carril: Este sistema es capaz de avisar al conductor si está sobrepasando las líneas de su carril sin activar el intermitente. Las principales tecnologías empleadas para esta función son tanto la imagen como el láser.
- Asistente de mantenimiento de carril: Este sistema se encarga de mantener el vehículo dentro del carril actuando directamente sobre la dirección de este para evitar salidas involuntarias. Las principales tecnologías empleadas para esta función son tanto la imagen como el láser. [6]

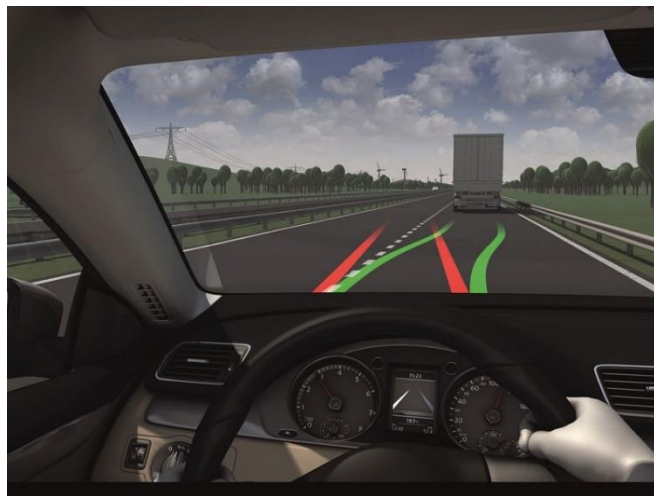


Ilustración 7. ADAS de Cambio involuntario de Carril. (Fuente: www.autopista.es)

- Colisión frontal: este sistema tiene impacto positivo en los accidentes en los que se produce una colisión frontal. Al igual que el sistema anterior, tiene dos niveles: el primero en el que el conductor simplemente recibe un aviso; y un segundo en el que el vehículo ya frena de forma automática.
 - Alerta de colisión frontal: Este sistema avisa al conductor si se dan situaciones en las que el riesgo de una colisión con un vehículo situado delante del propio sea elevado. Las principales tecnologías empleadas para esta función son tanto la imagen como radar.
 - Frenado automático de emergencia: Este sistema activa automáticamente los frenos del vehículo o aumento la presión del frenado si se detecta riesgo de colisión. Las principales tecnologías empleadas para esta función son tanto la imagen como el radar. [6]



Ilustración 8. ADAS de previsión de colisión frontal. (Fuente: www.autopista.es)

- Atropello de peatones y ciclistas: este sistema tiene también dos niveles de actuación: en primer lugar, el conductor simplemente recibe una alerta; y en segundo el vehículo activa autónomamente los frenos para evitar la colisión. El sistema funciona solo a baja velocidad y durante el día, pues necesita de buena visibilidad para un funcionamiento correcto. La tecnología empleada principalmente para esto es la imagen.
- Reconocimiento de señales de tráfico: este sistema reconoce las principales señales de tráfico que pueda haber en la vía, e indica de la existencia de estas al conductor. Puede mostrar la velocidad máxima de cada tramo, así como las prohibiciones. La tecnología empleada para ello es la imagen.



Ilustración 9. ADAS de previsión de colisión frontal. (Fuente: www.autopista.es)

2.2 Reconocimiento del estado de ánimo

El reconocimiento de emociones es el proceso de identificar emociones humanas. Esto es algo que el ser humano puede realizar de forma prácticamente automática, pero computacionalmente no es tan directo. La forma más típica de hacerlo es a partir de la expresión facial, pero se puede usar más información: hay proyectos que también emplean el tono de la voz e incluso el ritmo cardíaco. [7]

El reconocimiento del estado de ánimo, por tanto, implica una combinación de psicología y tecnología. La mayoría de herramientas que tratan de detectar el estado de ánimo humano lo suelen clasificar principalmente en siete categorías: alegría, tristeza, enfado, miedo, sorpresa, desprecio y neutral. [8]

Cada vez son más los proyectos que nacen, que se sirven reconocimiento del estado de ánimo con algún fin, la mayoría de ellos comerciales, pero también se puede llegar a usar en educación, sanidad y seguridad.

Algunos de estos proyectos se exponen a continuación:

2.2.1 Bismart Face and Emotion Recognition

Bismart es una empresa dedicada principalmente a la realización de proyectos de *Big Data*, *Machine Learning* e *Internet of Things* (IoT); siendo el principal socio de Microsoft en *Business Intelligence*. Entre sus proyectos, cabe destacar principalmente tres en los que integra visión artificial. En estos proyectos detecta e identifica rostros, además de reconocer sus estados de ánimo: [10]

- *Magic mirror*: Es un espejo inteligente que escanea la cara de un cliente, reconoce sus emociones y presenta las mejores opciones de compra adaptadas a él. Este proyecto permite obtener *feedback* en tiempo real sobre las emociones que los clientes experimentan mientras se ven con algún artículo que deseen comprar. Se van recogiendo las emociones y el sistema acaba recomendando el mejor artículo de acuerdo al estado emocional del cliente. [9]



Ilustración 10. Bismart Magic Mirror. (Fuente: www.bismart.com)

- *Happiness Recognition & Tourist Recommendations*: este proyecto realiza recomendaciones y sugerencias turísticas en función del estado emocional. [9]
- *Smart car safety system*: este sistema integrado en vehículos trata de aportar seguridad y comodidad en el vehículo. Si el sistema detecta que el conductor tiene sueño activa una alarma; por otro lado, si detecta que no es un conductor autorizado bloquea el motor de arranque. Además, permite personalizar los ajustes de asiento y de retrovisor para el conductor. [9]

2.2.2 Afectiva

Afectiva es una compañía dedicada fundamentalmente al desarrollo de tecnologías de reconocimiento de emociones. Nació de un proyecto del *MIT Media Lab* y es, posiblemente, la tecnología de este tipo más extendida. Trata de emplear tanto la imagen como las señales acústicas para determinar el estado de ánimo de la forma más precisa posible. [12 y 13]

El algoritmo empleado ha sido entrenado con la base de datos de imágenes de rostros más extensa de las empresas del sector: alrededor de 6 millones de rostros de 87 países diferentes. [12 y 13]

Los principales proyectos y productos a destacar de la compañía son:

- *Affective Automotive AI*: este proyecto es una inteligencia artificial que detecta a través de la imagen de la cara y del sonido de la voz el estado de ánimo tanto del conductor de un vehículo como de los pasajeros, de forma que se pueda utilizar esta información para mejorar tanto la seguridad como la experiencia en el transporte. [11]

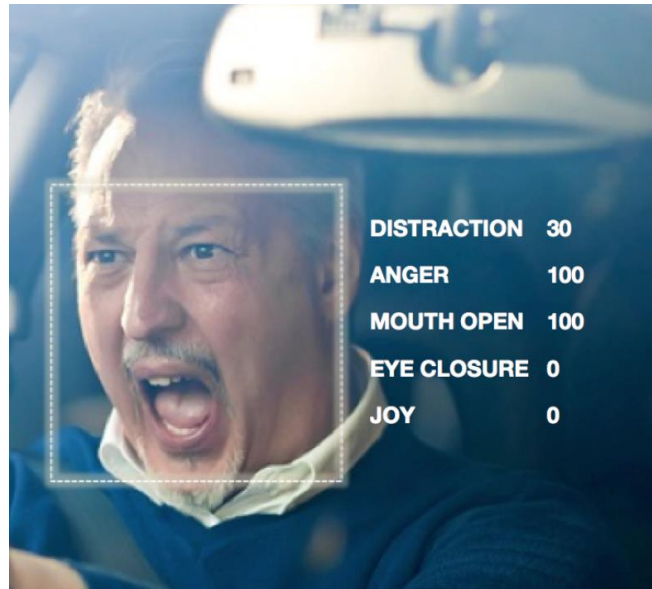


Ilustración 11. Affective Automotive AI. (Fuente: www.affective.com)

- *Affdex for Market Research*: este proyecto reconoce las reacciones emocionales del público con anuncios y otros contenidos digitales y las analiza. El fin de esto es optimizar el contenido digital y adaptar este a cada cliente. [11]
- *Emotion SDK*: es el SDK para incorporar la capa de reconocimiento de emociones de Affective a cualquier aplicación privada, de forma que los desarrolladores de cualquier aplicación puedan incorporar una capa de detección del estado de ánimo. [11]

3. Análisis del sistema

En este apartado se lleva a cabo un análisis de cómo debe ser el sistema. Para ello se realiza una especificación de requisitos tanto software como de usuario que deberán ser tenidos en cuenta en la implementación del proyecto.

Se realiza, también, una especificación de casos de uso en los que se exponen distintas situaciones de funcionamiento del sistema. Por último, se expone el marco regulador a tener en cuenta para su futura incorporación al mundo real.

3.1 Requisitos

Para la definición de los requisitos software del sistema se seguirá el siguiente modelo de tabla:

IDENTIFICADOR		TIPO_REQ-XX
Título	Titulo identificativo	
Necesidad	(3) Esencial / (2) Deseable / (1) Opcional	
Estabilidad	(3) Alta / (2) Media / (1) Baja	
Prioridad	(3) Alta / (2) Media/ (1) Baja	
Descripción	Descripción del requisito.	

Tabla 1. Tabla de especificación de Requisitos.

Donde cada campo indicará lo siguiente:

- **Identificador:** Cada requisito estará identificado por un código inicial (TIPO_REQ) que identificará el tipo de requisito. Habrá dos tipos principales, que a la vez se dividirán en otros dos subgrupos:
 - **RU:** Requisitos de usuario. Que se dividen en:
 - **RUC:** Requisitos de usuario de capacidad.
 - **RUR:** Requisitos de usuario de restricción.

○ **RS:** Requisitos software. Que se dividen en:

- **RSF:** Requisitos software funcionales.
- **RSNF:** Requisitos software no funcionales.

Además "XX" identificará el número del requisito en función de su tipo, siendo "01" el primero y "99" el último.

- **Título:** breve encabezado que identifique inequívocamente el propósito del requisito.
- **Necesidad:** La necesidad irá representada por un número, siendo 1 el más bajo, que representará un requisito opcional; y 3 el más alto, que pertenecerá a un requisito esencial para el usuario, siendo imprescindible su inclusión en el sistema.
- **Estabilidad:** Indicará el periodo de vida del requisito en la aplicación. El objetivo será que cuanto más estable sea el requisito mejor.
- **Prioridad:** Este campo contendrá el nivel de prioridad del requisito de cara a su implementación posterior en el sistema. Puede tomar tres valores posibles: siendo 1 el nivel prioritario más bajo, 2 el intermedio y 3 el nivel más alto.
- **Descripción:** Contendrá una explicación del requisito, que reflejará una cualidad o funcionalidad que el usuario desea o puede desear en algún momento.

3.1.1 Requisitos de usuario

Los requisitos de usuario son proporcionados por el cliente. Este, por medio de un lenguaje natural expone que desea que realice el sistema y los servicios que debe proporcionar, además de las restricciones de operación con las que debe contar [14].

A continuación, se exponen los requisitos de usuario del sistema, tanto de capacidad como de restricción:

3.1.1.1. Requisitos de capacidad

Los requisitos de capacidad describen servicios que el cliente determina que debe poder llevar a cabo el sistema. Se exponen a continuación:

IDENTIFICADOR RUC-01	
Título	Detección de rostro
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	El sistema de ser capaz de identificar rostros.

Tabla 2. Requisito de Capacidad RUC-01

IDENTIFICADOR RUC-02	
Título	Facial Landmarks
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	El sistema de ser capaz de identificar los puntos más característicos de la cara

Tabla 3. Requisito de Capacidad RUC-02

IDENTIFICADOR RUC-03	
Título	Identificación del estado de ánimo
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	El sistema de poder reconocer el estado de ánimo de un rostro identificado.

Tabla 4. Requisito de Capacidad RUC-03

IDENTIFICADOR RUC-04	
Título	Modo debug
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	El sistema podrá ser ejecutado en modo debug para su desarrollo y depurar errores.

Tabla 5. Requisito de Capacidad RUC-04

IDENTIFICADOR RUC-05	
Título	Identificación del rostro en modo debug
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	Cuando el sistema se ejecute en modo debug, se mostrará la secuencia de imágenes en tiempo real en la que se pueda ver el rostro identificado, en caso de haberlo hecho; y sus puntos más característicos.

Tabla 6. Requisito de Capacidad RUC-05

IDENTIFICADOR RUC-06	
Título	Estado de ánimo en modo debug
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	Cuando se ejecute el sistema en modo debug también se mostrará el estado de ánimo del rostro identificado.

Tabla 7. Requisito de Capacidad RUC-06

IDENTIFICADOR RUC-07	
Título	Registro de funcionamiento del sistema
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	Se generará un registro con todas las operaciones llevadas a cabo por el sistema cada vez que este se ejecute.

Tabla 8. Requisito de Capacidad RUC-07

IDENTIFICADOR RUC-08	
Título	Entrenamiento del modelo predictivo
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	Se podrán generar diferentes modelos predictivos para determinar el estado de ánimo del conductor durante el ciclo de vida del sistema.

Tabla 9. Requisito de Capacidad RUC-08

IDENTIFICADOR RUC-09	
Título	Aviso de errores
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	El sistema emitirá una señal cuando se produzca un error en tiempo de ejecución, que, además, quedará reflejada en el registro de funcionamiento.

Tabla 10. Requisito de Capacidad RUC-09

3.1.1.2. Requisitos de Restricción

Los requisitos de capacidad describen servicios que el cliente determina que debe poder llevar a cabo el sistema. Se exponen a continuación:

IDENTIFICADOR		RUR-01
Título		Posición de la cámara
Necesidad		3
Estabilidad		3
Prioridad		3
Descripción		El rostro del usuario debe estar en frente de la cámara para determinar su estado de ánimo.

Tabla 11. Requisito de Restricción RUR-01

IDENTIFICADOR		RUR-02
Título		Conexión a internet
Necesidad		3
Estabilidad		3
Prioridad		3
Descripción		El sistema podrá funcionar tanto si se cuenta con conexión a internet como si no.

Tabla 12. Requisito de Restricción RUR-02

IDENTIFICADOR		RUR-03
Título		Formato del registro de ejecución
Necesidad		3
Estabilidad		3
Prioridad		3
Descripción		El registro de ejecución se guardará en formato <i>.txt</i> .

Tabla 13. Requisito de Restricción RUR-03

IDENTIFICADOR RUR-04	
Título	Estados de ánimo
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	El sistema reconocerá los siguientes estados de ánimo: neutral, contento, triste, sorprendido, enfadado y angustiado.

Tabla 14. Requisito de Restricción RUR-04

IDENTIFICADOR RUR-05	
Título	Imágenes de entrenamiento
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	El sistema debe disponer al menos de 20 imágenes de muestra diferente de cada estado de ánimo para entrenar un nuevo modelo predictivo.

Tabla 15. Requisito de Restricción RUR-05

IDENTIFICADOR RUR-06	
Título	Actualización del estado de ánimo
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	El estado de ánimo será comprobado al menos una vez por segundo y actualizado en caso de ser necesario.

Tabla 16. Requisito de Restricción RUR-06

3.1.2 Requisitos Software

Los requisitos software del sistema describen las funcionalidades, servicios y restricciones del sistema de una manera más detallada y técnica; y son determinados tanto por el cliente como por el contratante [14].

A continuación, se lleva a cabo la especificación de requisitos software del sistema, tanto funcionales como no funcionales:

3.1.3.1. Requisitos funcionales

Cada requisito funcional representa una funcionalidad que realiza el software del sistema. A continuación, se exponen los requisitos funcionales del sistema:

IDENTIFICADOR		RSF-01
Título		Detección del rostro
Necesidad		3
Estabilidad		3
Prioridad		3
Descripción		El sistema será capaz de detectar el rostro del conductor de forma continua.

Tabla 17. Requisito Funcional RSF-01

IDENTIFICADOR		RSF-02
Título		Detección de Landmarks
Necesidad		3
Estabilidad		3
Prioridad		3
Descripción		El sistema será capaz detectar los <i>Facial Landmarks</i> en el rostro del conductor de forma continua.

Tabla 18. Requisito Funcional RSF-02

IDENTIFICADOR RSF-03	
Título	Detección de la inclinación de la cara
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	El sistema será capaz de detectar si la cara del conductor esta inclinada.

Tabla 19. Requisito Funcional RSF-03

IDENTIFICADOR RSF-04	
Título	Corrección de la inclinación
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	El sistema será capaz de corregir la inclinación detectada de la cara del conductor.

Tabla 20. Requisito Funcional RSF-04

IDENTIFICADOR RSF-05	
Título	Detección del punto medio
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	El sistema detectará el punto medio de la cara del conductor.

Tabla 21. Requisito Funcional RSF-05

IDENTIFICADOR RSF-06	
Título	Activación interfaz gráfica
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	El sistema tendrá una opción en la que un administrador active la interfaz gráfica.

Tabla 22. Requisito Funcional RSF-06

IDENTIFICADOR RSF-07	
Título	Dibujo de puntos de interés
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	El sistema dibujará todos los puntos de interés (Landmarks y punto medio de la cara) en la imagen del rostro cuando se active la interfaz gráfica.

Tabla 23. Requisito Funcional RSF-07

IDENTIFICADOR RSF-08	
Título	Detección del estado de ánimo
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	El sistema será capaz de detectar el estado de ánimo del conductor de forma continua.

Tabla 24. Requisito Funcional RSF-08

IDENTIFICADOR RSF-09	
Título	Modo depuración o debug
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	El sistema podrá ejecutarse en modo debug.

Tabla 25. Requisito Funcional RSF-09

IDENTIFICADOR RSF-10	
Título	Registro de funcionamiento
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	El sistema escribirá un log en el que aparecerá cada cambio de estado de ánimo, así como la información de interés para detectar el cambio.

Tabla 26. Requisito Funcional RSF-10

IDENTIFICADOR RSF-11	
Título	Reentrenamiento del sistema
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	Un administrador podrá crear un nuevo modelo de predicción del estado de ánimo para ser usado por el sistema.

Tabla 27. Requisito Funcional RSF-11

IDENTIFICADOR RSF-12	
Título	Acierto del modelo
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	El sistema mostrará el porcentaje del modelo predictivo cuando este sea entrenado.

Tabla 28. Requisito Funcional RSF-12

IDENTIFICADOR RSF-13	
Título	Funcionamiento de la cámara
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	El sistema indicará cuando la cámara está desconectada o no funciona correctamente.

Tabla 29. Requisito Funcional RSF-13

IDENTIFICADOR RSF-14	
Título	Reporte de errores
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	El sistema lanzará una excepción cuando ocurra un error durante la ejecución y será anotado en el log de ejecución.

Tabla 30. Requisito Funcional RSF-14

IDENTIFICADOR RSF-15	
Título	Control de la cámara
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	El sistema tendrá el control de una cámara para la extracción de imágenes.

Tabla 31. Requisito Funcional RSF-15

IDENTIFICADOR RSF-16	
Título	Dibujo de la localización facial
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	El sistema dibujará un rectángulo en el área en el que se detectó el rostro cuando el modo de interfaz gráfica esté activado.

Tabla 32. Requisito Funcional RSF-16

IDENTIFICADOR RSF-17	
Título	Dibujo del punto medio
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	El sistema dibujará el punto medio del rostro cuando el modo de interfaz gráfica esté activado.

Tabla 33. Requisito Funcional RSF-17

IDENTIFICADOR RSF-18	
Título	Estado de ánimo con interfaz gráfica
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	El sistema indicará al lado del rostro detectado su estado de ánimo cuando el modo de interfaz gráfica esté activado.

Tabla 34. Requisito Funcional RSF-18

IDENTIFICADOR		RSF-19
Título	Interfaz gráfica en modo debug	
Necesidad	3	
Estabilidad	3	
Prioridad	3	
Descripción	El sistema iniciará la interfaz gráfica cuando se ejecute en modo debug.	

Tabla 35. Requisito Funcional RSF-19

3.1.3.2. Requisitos no funcionales

Cada requisito no funcional representa una característica del funcionamiento del sistema, y no una funcionalidad. A continuación, se exponen los requisitos no funcionales del sistema:

IDENTIFICADOR		RSNF-01
Título		Posición de la cabeza
Necesidad		3
Estabilidad		3
Prioridad		3
Descripción		La cabeza del conductor debe estar situada a una distancia mínima de 40cm de la cámara, mirando a ésta de frente.

Tabla 36. Requisito no Funcional RSNF-01

IDENTIFICADOR		RSNF-02
Título		Posición de la cámara
Necesidad		3
Estabilidad		3
Prioridad		3
Descripción		La cámara estará situada en el techo del coche, a la altura del volante.

Tabla 37. Requisito no Funcional RSNF-02

IDENTIFICADOR		RSNF-03
Título		Situación del rostro
Necesidad		3
Estabilidad		3
Prioridad		3
Descripción		La situación del rostro del conductor será indicada por 4 puntos, que corresponderán con las esquinas de un rectángulo.

Tabla 38. Requisito no Funcional RSNF-03

IDENTIFICADOR	RSNF-04
Título	Formato del log de ejecución
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	El log de ejecución se guardará en formato <i>.txt</i>

Tabla 39. Requisito no Funcional RSNF-04

IDENTIFICADOR	RSNF-05
Título	Frames por segundo
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	El flujo de imágenes que se obtenga de la cámara debe de ser de al menos 5 frames por segundo.

Tabla 40. Requisito no Funcional RSNF-05

IDENTIFICADOR	RSNF-06
Título	Cantidad de imágenes de entrenamiento
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	Se deben tener al menos 20 imágenes de cada categoría de estado de ánimo para entrenar un nuevo modelo

Tabla 41. Requisito no Funcional RSNF-06

IDENTIFICADOR	RSNF-07
Título	Características de imágenes de entrenamiento
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	Las imágenes de entrenamiento deben mostrar la cara de frente, en formato <i>.png</i> , y de resolución mínima 640x490 px.

Tabla 42. Requisito no Funcional RSNF-07

IDENTIFICADOR RSNF-08	
Título	Estados de ánimo
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	Los estados de ánimo del sistema son : neutral, contento, triste, sorprendido, enfadado y angustiado.

Tabla 43. Requisito no Funcional RSNF-08

IDENTIFICADOR RSNF-09	
Título	Conexión a internet
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	El sistema podrá funcionar con conexión a internet.

Tabla 44. Requisito no Funcional RSNF-09

IDENTIFICADOR RSNF-10	
Título	Funcionamiento offline
Necesidad	3
Estabilidad	3
Prioridad	3
Descripción	El sistema podrá funcionar sin conexión a internet.

Tabla 45. Requisito no Funcional RSNF-10

3.2 Casos de uso

Para la descripción de los casos de uso se utilizará la siguiente plantilla:

IDENTIFICADOR	CU-XX
Título	
Objetivo	
Precondiciones	
Postcondiciones	
Escenario de éxito	

Tabla 46. Casos de uso.

Donde cada campo indicará lo siguiente:

- **Identificador:** Cada caso de uso estará identificado por las siglas CU (Caso de Uso). "XX" identificará el número del caso de uso, siendo "01" el primero y "99" el último.
- **Título:** Breve encabezado que identifique inequívocamente el propósito del caso de uso.
- **Precondiciones:** Condiciones que se deben cumplir para llevar a cabo la operación.
- **Postcondiciones:** Estado del sistema tras realizar la operación
- **Escenario de éxito:** Conjunto de eventos que dan lugar a la consecución del caso de uso.

Los casos de uso se exponen a continuación:

IDENTIFICADOR CU-01	
Título	Entrenamiento del sistema
Objetivo	Crear un modelo de predicción que permita, posteriormente, indicar el estado de ánimo a partir de la imagen de un rostro.
Precondiciones	<ul style="list-style-type: none"> ▪ Disponer de un base de datos de imágenes de rostros. ▪ Clasificar la base de datos en los estados de ánimo de los que se dispone y que se desea predecir.
Postcondiciones	<ul style="list-style-type: none"> ▪ Se genera un modelo predictivo y se muestra su porcentaje de acierto.
Escenario de éxito	<ul style="list-style-type: none"> ▪ Ejecutar el módulo de entrenamiento en la parte del computador que haga de servidor. ▪ Se genera el archivo que contiene el modelo predictivo y se almacena en el computador que haga de servidor. ▪ La parte del servidor carga el archivo que contiene el modelo predictivo.

Tabla 47. Caso de Uso CU-01

IDENTIFICADOR CU-02	
Título	Conexión con el servidor
Objetivo	Conectar las aplicaciones que hacen de cliente y servidor para permitir el flujo de información entre ambas.
Precondiciones	<ul style="list-style-type: none"> ▪ Debe haberse generado un modelo predictivo. ▪ El programa servidor que determinará el estado de ánimo debe estar ejecutándose ▪ El programa cliente que controlará la cámara debe ejecutarse.
Postcondiciones	<ul style="list-style-type: none"> ▪ El servidor indica al programa cliente el estado de ánimo del conductor.
Escenario de éxito	<ul style="list-style-type: none"> ▪ Ejecutar la aplicación que hace de servidor. ▪ Se carga el modelo predictivo correctamente. ▪ Ejecutar la aplicación que hace de cliente. ▪ Se establece la conexión entre ambas partes del sistema. ▪ Se produce un intercambio de información.

Tabla 48. Caso de Uso CU-02

IDENTIFICADOR CU-03	
Título	Detección facial
Objetivo	La detección facial del usuario conductor del vehículo para la obtención posterior de información.
Precondiciones	<ul style="list-style-type: none"> ▪ Debe haberse ejecutado el programa cliente. ▪ La cámara ha sido reconocida e iniciada correctamente. ▪ El usuario debe estar situado en una posición correcta.
Postcondiciones	<ul style="list-style-type: none"> ▪ El programa cliente detecta la posición de la cara del usuario.
Escenario de éxito	<ul style="list-style-type: none"> ▪ Ejecutar la aplicación ▪ Activarse la cámara ▪ Estar en una posición correcta. ▪ Se obtienen cuatro puntos que determinan la posición de la cara.

Tabla 49. Caso de Uso CU-03

IDENTIFICADOR CU-04	
Título	Detección de Landmarks
Objetivo	Obtención de los <i>Facial Landmarks</i> de la imagen del rostro del usuario, así como información importante a partir de estos puntos.
Precondiciones	<ul style="list-style-type: none"> ▪ Debe haberse detectado la posición del rostro del usuario en la imagen.
Postcondiciones	<ul style="list-style-type: none"> ▪ Se obtienen todos los <i>Facial Landmarks</i> de la imagen del rostro del usuario ▪ Se obtiene el punto medio de la cara del usuario ▪ Se obtiene la inclinación de la cara del usuario.
Escenario de éxito	<ul style="list-style-type: none"> ▪ Se ha detectado el rostro del conductor. ▪ Se obtienen los <i>Facial Landmarks</i> y toda la información de interés.

Tabla 50. Caso de Uso CU-04

IDENTIFICADOR		CU-05
Título	Detección del estado de ánimo	
Objetivo	Se obtiene el estado de ánimo del conductor del usuario.	
Precondiciones	<ul style="list-style-type: none"> ▪ Debe haberse obtenido toda la información necesaria de la imagen del rostro del usuario ▪ Se ha establecido la conexión entre los programas cliente y servidor. 	
Postcondiciones	<ul style="list-style-type: none"> ▪ El servidor indica al programa cliente el estado de ánimo del usuario. 	
Escenario de éxito	<ul style="list-style-type: none"> ▪ Se han conectado correctamente cliente y servidor ▪ Se ha obtenido toda la información necesaria de la imagen del rostro del conductor. ▪ El servidor determina el estado de ánimo del conductor ▪ Se activa una alarma en el caso correspondiente. 	

Tabla 51. Caso de Uso CU-05

3.3 Marco regulador

El proyecto se base en monitorizar el rostro del conductor de un vehículo, por lo que al estar grabándole se usan continuamente datos personales de este. De esta forma, es necesario cumplir la Ley Orgánica 15/1999 de Protección de Datos de Carácter Personal. [15]

Por otro lado, el sistema ha sido creado con la intención de ser incorporado en los vehículos que circulan por las vías con normalidad, pudiendo llegar a tomar control en este en un futuro. Esto requiere la Autorización de pruebas o ensayos de investigación realizados con vehículos de conducción automatizada en vías abiertas al tráfico en general, que se encuentra en la Instrucción 15/V-113 de la DGT. [16]

4. Arquitectura y Diseño

En esta sección se lleva a cabo una descripción del sistema, incluyendo los tres principales módulos que van a componer éste, indicando las principales funciones de cada uno y la interacción entre ellos.

Además, en el apartado 4.1 *Estudio de la solución final* se exponen distintas alternativas a tener en cuenta para el desarrollo del sistema. La solución escogida a dichas alternativas, se expone en el subapartado 4.1.1 *Solución elegida*.

4.1 Arquitectura

El sistema estará compuesto por tres módulos. Cada uno de ellos realiza principalmente una función y la comunicación entre ellos se realizará siempre a partir del *módulo servidor*, que será uno de estos módulos. Los otros dos módulos serán el *módulo de entrenamiento* y el *módulo cliente*.

A continuación, se expone un diagrama en el que se muestran los módulos y sus relaciones:

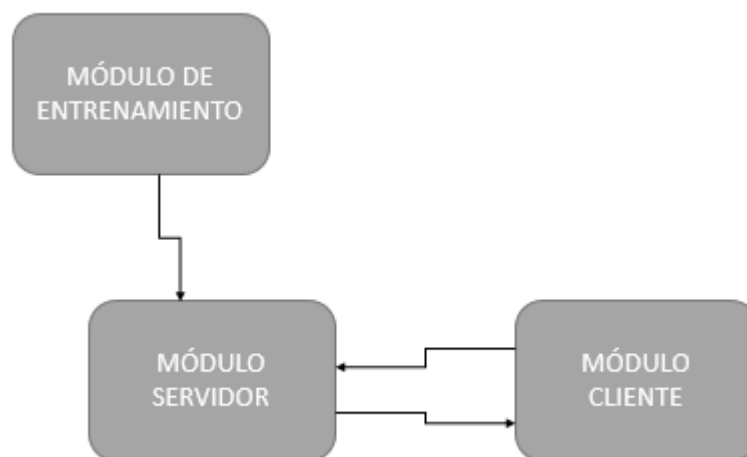


Ilustración 12. Diagrama de arquitectura del sistema.

4.1.1 Módulo de entrenamiento

Esta parte del sistema contiene el algoritmo que posteriormente será usado para extraer información de las imágenes de los rostros de los usuarios.

Requiere la existencia de una base de datos de rostros clasificada por estados de ánimo. En el apartado de requisitos se menciona como debe ser dicha base de datos.

A partir del algoritmo y la base de datos, el módulo de entrenamiento es capaz de generar un modelo predictivo que posteriormente será usado por el módulo servidor para determinar el estado de ánimo de los usuarios.

4.1.2 Módulo Cliente

La principal función de esta parte del sistema es la del manejo de la cámara. Inicia la cámara y extrae un flujo continuo de imágenes. Si se detecta un rostro en la imagen, recorta el área de este, lo envía al módulo servidor y espera la respuesta de este. La respuesta contendrá el estado del ánimo del rostro que fue detectado.

Este módulo, además, controla los posibles errores de ejecución que puedan tener que ver con la cámara producidos en tiempo de ejecución, o en caso de no estar conectada.

4.1.3 Modulo Servidor

Este módulo carga, en primer lugar, el modelo predictivo generado por el módulo de entrenamiento. Si no se ha generado dicho modelo, no podrá ejecutarse.

Una vez cargado el modelo, espera a recibir una imagen de un rostro del módulo cliente. Cuando recibe esta imagen extrae toda la información necesaria del rostro empleando el mismo algoritmo empleado en el módulo de entrenamiento y predice el estado de ánimo del usuario.

El estado de ánimo es la respuesta que se le envía al módulo cliente.

4.2 Estudio de la solución final

Como se muestra en la *Ilustración 3. Diagrama de la arquitectura del sistema*, el sistema está formado por tres módulos:

- Módulo de entrenamiento, que permitirá entrenar un modelo de predicción.
- Módulo servidor, que realizará todas las operaciones principales para obtener el estado de ánimo de un rostro.
- Módulo cliente, que su función principal será controlar la cámara y obtener un rostro.

En primer lugar, era necesario un módulo específico para generar un modelo predictivo. Entrenar un modelo lleva gran cantidad de tiempo, por lo que realizar esta operación cada vez que se ejecute el sistema, generando siempre el mismo modelo no tiene sentido; sobre todo cuando se puede generar y guardar uno para las ejecuciones posteriores.

Además, disponer del módulo de entrenamiento separado de los demás, también permite modificar el modelo predictivo para actualizaciones futuras del software. Eso significa que se puede actualizar tanto el algoritmo usado para entrenar el modelo, como la base de datos de rostros, como las características que se extraen de cada rostro. Debe tenerse en cuenta que si se modifican las características que se extraen de la imagen de un rostro, debe actualizarse también el módulo servidor para que extraiga las mismas y, de este modo, el nuevo modelo predictivo pueda funcionar y determinar el estado de ánimo.

Por otro lado, si los módulos servidor y cliente fueran un solo módulo, todas las operaciones se realizarían de forma secuencial. Sin embargo, con un modelo cliente-servidor se dispone de las siguientes ventajas, que han motivado a elegir esta arquitectura:

Hoy día, cada vez más automóviles disponen de conectividad a internet. Esto permite que el módulo servidor, que realiza la mayor parte de operaciones, se encuentre en un dispositivo con mayor capacidad de cómputo a la que pueda tener un automóvil. Reduciendo a su vez, la cantidad de procesamiento que debería realizar este o los costes que conllevaría una unidad con mayor capacidad de cómputo en el vehículo.

La transmisión de información es de pequeños bloques, por lo que estas operaciones con buena conectividad llevarán poco tiempo.

Esto, además, permite que el computador en el que se aloje el servidor reciba peticiones de diversos automóviles.

Por último, si se deseara actualizar el modelo predictivo (como se ha mencionado anteriormente), solo habría que actualizar el servidor, ya fuera cargando el nuevo modelo o actualizando el algoritmo. Y, si el servidor se encontrase en un computador externo y no en el del propio vehículo, no implicaría la necesidad de producir una actualización que instalar en el computador del propio vehículo, de forma que no afectaría al usuario final.

5. Implementación y despliegue

En esta sección se trata la implementación y el despliegue del sistema. En primer lugar, se describirá la implementación, justificando las decisiones tomadas en este proceso. Después, se indicarán los puntos a tener en cuenta para su despliegue y correcto funcionamiento.

5.1 Implementación

Como se ha mencionado anteriormente, para comenzar con la implementación, se debía encontrar una base de datos de imágenes de rostros clasificada por estados de ánimo. Sin ella, no habría posibilidad de entrenar un modelo predictivo, que es algo fundamental para el funcionamiento del proyecto.

Tras esto, es necesario que el computador pueda controlar y operar sobre las imágenes. Para esto se ha empleado una librería de Visión Artificial que da gran soporte para estas tareas: *OpenCV*. [17]

Lo siguiente es detectar los rostros y decidir qué información se va a analizar de ellos. Esta es la parte más importante a tener en cuenta para una correcta detección del estado de ánimo. Analizar demasiada información, pues si no el sistema tendría un funcionamiento demasiado lento. Se trata de analizar la información mínima pero más descriptiva para determinar el estado de ánimo de un rostro. Por esto, la conclusión ha sido obtener los *Facial Landmarks*. Más adelante se explicará el por qué. [18]

Con las imágenes y la información de ellas, ya se puede generar un modelo predictivo. Se han hecho pruebas con más y menos información recogida y con distintos algoritmos de clasificación supervisados.

Una vez se dispone del modelo predictivo, ya se puede clasificar imágenes de rostros obtenidos a partir de una cámara. La cámara, cuya disposición se menciona en el apartado de requisitos, será la fuente de información para determinar el estado de ánimo del conductor.

Por último, se explicará el funcionamiento de los módulos cliente y servidor. Como ya se menciona anteriormente, el cliente controla la cámara en tiempo de ejecución y el servidor obtiene el estado de ánimo. Se explicará cómo se realiza la interconexión entre ambos y como es el intercambio de datos.

5.1.1 Generar modelo predictivo

Como se ha mencionado anteriormente, para generar un modelo predictivo es necesario disponer de una base de datos con rostros clasificada por estados de ánimo. La base de datos empleada ha sido la base de datos extendida de *Cohn-Kanade*. Esta base de datos contiene imágenes de rostros de 122 sujetos con 8 estados de ánimo, dando lugar un total de casi 500 imágenes de muestra. Los estados de ánimo que contiene esta base de dato son enfado, disgusto, susto, alegría, tristeza, sorpresa, desprecio y neutral. En las siguientes ilustraciones se muestran imágenes de muestra de todas estas categorías: [19]



Ilustración 13. Imágenes de muestra de las categorías empleadas de la base de datos. De izquierda a derecha: enfado, disgusto y susto en la primera fila; alegría, tristeza y sorpresa en la segunda.



Ilustración 14. Imágenes de muestra de las categorías empleadas de la base de datos. De izquierda a derecha: neutral y desprecio.

De estas 8 categorías diferentes, se ha tenido que descartar de entrada la categoría de desprecio, ya que, como se especificaba en la sección de requisitos, carecía de muestras suficientes para entrenar el sistema.

Por otro lado, cabe destacar que finalmente se han utilizado las 7 categorías restantes; pero dos de ellas, enfado y disgusto, se ha optado por unirlos como una sola categoría por motivos que se explicarán más adelante en este mismo apartado.

Una vez se dispone de una base de datos de imágenes de rostros clasificada por estados de ánimo con muestras suficientes, el siguiente paso es reconocer el rostro en cada imagen. Para esta tarea se emplean funciones propias de las librerías de visión artificial y aprendizaje automático *OpenCV* y *dLib*, ya que ofrecen gran soporte para estas tareas.

Cuando se detecta un rostro en una imagen, se obtienen cuatro puntos que forman un rectángulo determinando la posición en la que el rostro ha sido encontrado.



Ilustración 15. Imagen de muestra para detectar un rostro.



Ilustración 16. Imagen de muestra con rostro detectado.

Una vez se ha identificado la posición del rostro dentro de la imagen, ya se puede extraer información de este.

Cabe destacar que tanto las características que se extraen de cada imagen como el algoritmo empleado de clasificación afectan directamente a la probabilidad de acierto en la predicción del

estado de ánimo; de modo que esta es la parte más importante en el desarrollo para lograr el mejor comportamiento del sistema posible.

Mencionar, además, que tanto el algoritmo de clasificación como las características afectarán también al tiempo requerido para entrenar el modelo, como el tiempo necesario para realizar la predicción.

Anteriormente, en el apartado de requisitos (*3.1 Requisitos*), fueron definidas las emociones a detectar, como debían ser las imágenes de entrenamiento y la cantidad de estas. A continuación, se exponen una serie de casos para la posterior realización de tests y determinar así qué algoritmo usar y qué características se deben tener en cuenta para entrenar el sistema con los mejores resultados posibles:

- Caso 1: Se entrena para reconocer los seis estados de ánimo más el estado neutral, se corrige la posible inclinación que tenga la cara y se utilizan los siguientes datos de cada rostro:
 - *Facial Landmarks*: sesenta y ocho puntos que delimitan los contornos de los elementos más importantes en la gesticulación facial: los primeros 17 puntos delimitan el contorno de la cara; del punto 18 al 22 la ceja derecha; del 23 al 27 la ceja izquierda; del 28 al 36 la nariz; del 37 al 42 al ojo derecho y del 43 al 48 el izquierdo; y, por último, los puntos del 49 al 68 delimitan lo labios.



Ilustración 17. Ejemplo de Facial Landmarks.

- Distancia de cada *Facial Landmark* al punto medio del rostro
- Inclinación que guarda cada *Facial Landmark* respecto al punto medio de la cara.
- Caso 2: En la matriz de confusión del caso anterior se puede observar se encuentran dificultades al tratar de diferenciar entre los estados enfadado y disgustado.

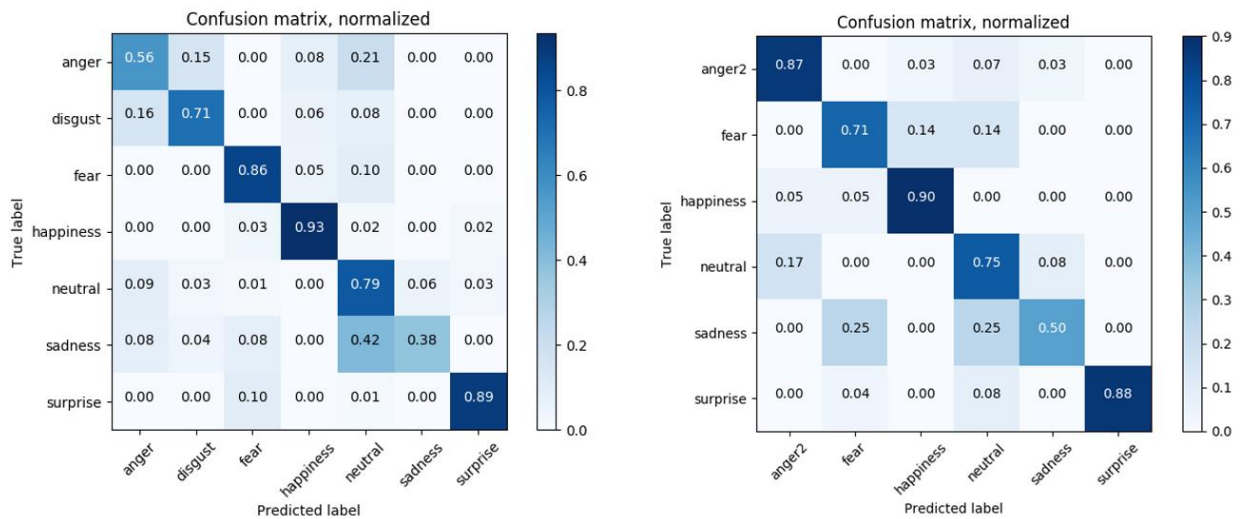


Ilustración 18. Matrices de confusión normalizadas para caso 1 y caso 2 para el algoritmo SVM de kernel lineal. Dónde: anger2 es la unión de enfado y disgusto, fear es miedo, happiness alegría, neutral es neutral, sadness tristeza, surprise sorpresa, anger enfado y disgust disgusto.

Como se ve en la imagen que aparece a continuación, ambos gestos son muy parecidos:



Ilustración 19. Imágenes de muestra: Enfadado y Disgustado.

Es por esto, que se ha decidido un segundo caso en ambos gestos se unifican en una sola categoría, que será “enfado2”.

Por otro lado, se emplean los mismos datos de entrenamiento que en el caso anterior.

- Caso 3: En el tercer caso se unen de nuevo las clases *Enfadado* y *Disgustado* formando una única clase, *Enfadado*; y se utilizan como datos para entrenar la distancia de los puntos al punto medio de la cara y el ángulo que forman con este punto, después de corregir la inclinación de la cara. Los *Facial Landmarks* no se usan de forma directa en este caso.
- Caso 4: En el tercer caso se unen de nuevo las clases *Enfadado* y *Disgustado* formando una única clase, *Enfadado*; y se utilizan como datos únicamente los *Facial Landmarks* en crudo.

Los algoritmos elegidos para hacer las pruebas son:

- Support Vector Machine (SVM) con la función del kernel lineal.
- Support Vector Machine (SVM) con la función del kernel polinomial.
- Random Forest

Los tres son algoritmos de clasificación supervisados. Esto significa que están definidas las clases en las que se dividen los conjuntos, que son los diferentes estados de ánimo.

El porcentaje de acierto de cada algoritmo elegido para cada caso expuesto se muestra a continuación:

	Caso 1	Caso 2	Caso 3	Caso 4
SVM Lineal	80 %	86 %	83 %	84 %
SVM Polinomial	70 %	76 %	80 %	81 %
Random Forest	72 %	81 %	75 %	61 %

Tabla 52. Tests de algoritmos.

En primer lugar, cabe destacar la mejora obtenida al unir las clases *Enfadado* y *Disgustado* en una sola. Ambas clases compartían características muy similares.

Por otro lado, destaca el resultado del *SVM Lineal* sobre los demás algoritmos. Sobre todo, en el que se usan todas las características, aunque no hay gran mejora respecto al que utiliza los *Facial Landmarks* únicamente. De esto se puede concluir que no hay un gran aporte de información extra y que estos puntos, por si solos ya dan información suficiente para generar un buen clasificador.

En el caso del *SVM Polinomial* se puede ver como se da el fenómeno de *overfitting*. Esto se produce por un sobreajuste a la hora de elegir los datos de entrenamiento, es decir, se usan demasiados datos de entrenamiento, lo que da lugar a una mala clasificación. [20]

Por último, mencionar que no hay mucho que destacar sobre el algoritmo *Random Forest*. Sus resultados son peores que los obtenidos con el *SVM Lineal*.

Con todo esto, modelo predictivo elegido para el sistema es el generado por el *SVM Lineal* en el caso 2, ya que produce los mejores resultados.

5.1.2 Funcionamiento del sistema

Una vez entrenado el modelo predictivo y guardado en un archivo, el funcionamiento del sistema es el siguiente:

En primer lugar, debe ejecutarse el módulo servidor. Este módulo funciona como una aplicación web. Primero cargará el modelo predictivo para después poder predecir estados de ánimo. El servidor, si se ejecuta en un computador externo, estará esperando continuamente a recibir peticiones de otros módulos clientes que puedan conectarse a él; por otro lado, si se ejecuta en local (en el propio vehículo), cliente y servidor se encontrarán ejecutando en la misma red y estará esperando hasta que el módulo cliente detecte el rostro del usuario conductor del vehículo.

Con el servidor correctamente iniciado y funcionando, la parte cliente ya puede iniciarse. Cuando ésta se inicia la primera acción que lleva a cabo es encender la cámara. La cámara comienza a tomar imágenes hasta que se detecta un rostro. Cuando se detecta el rostro del usuario, se recorta el área de la imagen en la que se encuentra el rostro y se le envía al servidor.

Cuando el servidor recibe una petición del cliente con la imagen de un rostro, extrae las mismas características que se extrajeron de cada rostro anteriormente para entrenar el modelo predictivo. Es decir, se extraen los *Facial Landmarks*, se corrige la inclinación de la cara y se guarda tanto la distancia de cada *Facial Landmark* como su inclinación respecto al punto medio de la cara.

Una vez extraídas todas las características descritas, el módulo servidor ya dispone de la información necesaria para que al modelo predictivo pueda determinar el estado de ánimo del rostro de la imagen recibida.

Por último, se le envía el resultado de la predicción a la parte cliente, de forma que pueda tomar alguna acción; y el servidor procesa los datos de ejecución en un log.

A continuación, se adjunta un diagrama que muestra el funcionamiento descrito del sistema:

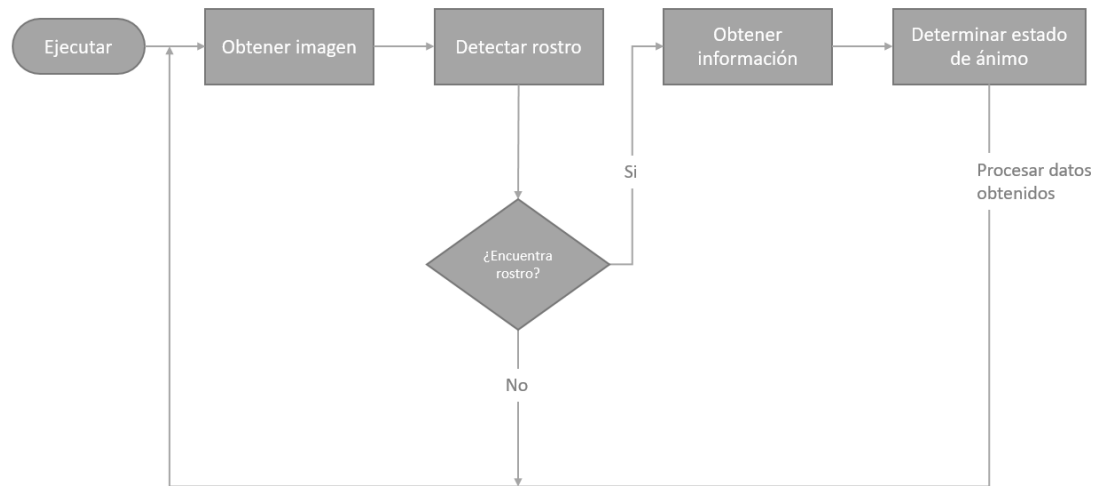


Ilustración 20. Diagrama de ejecución del sistema.

5.1.3 Cliente: manejo de la cámara

Anteriormente se mencionó que la principal función de esta parte del sistema era el manejo de la cámara y la extracción de fotogramas. Las acciones que debe realizar este módulo del sistema, por tanto, son:

- 1- Iniciar la cámara. Una vez se ejecuta la instrucción de para arrancar la cámara se esperan dos segundos y se comprueba que esta se ha iniciado. De no ser así, se lanza una excepción indicando que ha habido un error en el proceso de iniciar la cámara, ya sea por no estar ésta conectada o por otros motivos.
- 2- Una vez se ha iniciado correctamente la cámara, se comienza a obtener imágenes de forma continua. De cada imagen se obtiene una segunda copia en blanco y negro, que será la imagen con la que se lleven a cabo las principales operaciones del sistema. La imagen original, en color, se mantendrá para dibujar el resultado en caso de ser necesario o realizar alguna operación de mantenimiento.
- 3- Para cada imagen obtenida se intenta identificar un rostro. Si no se encuentra ningún rostro, se vuelve al segundo paso y se extrae una imagen nueva. De detectarse algún rostro se continuaría con el flujo de operaciones.
- 4- Una vez detectado el rostro, se recorta la imagen para poder enviar posteriormente la imagen del rostro al servidor.
- 5- Se envía la imagen del rostro recortado al servidor y se espera la respuesta de este.
- 6- Una vez recibida la respuesta ya el sistema ya sabe el estado de ánimo del conductor, por lo que volvería al punto dos en el que se extrae una nueva imagen del conductor.

Si se deseara, durante el desarrollo podría mostrarse el resultado de la ejecución con imágenes como la siguiente, en la que ya podría verse el estado de ánimo del conductor:



Ilustración 21. Resultado de ejecución.

5.1.4 Servidor

El módulo servidor del sistema se despliega bajo el *microframework Flask*. [23] Este *microframework* despliega la parte servidor de la aplicación como un servicio web, facilitando tanto el propio despliegue del módulo, como el manejo de peticiones recibidas de clientes. Por estos motivos se decidió usar este *microframework* y no otro.

Ya se mencionó anteriormente que la principal función del módulo era la extracción de información de las imágenes y la de determinar el estado de ánimo del rostro de esta imagen. Las operaciones que debe llevar a cabo el módulo para esto son:

- 1- Cargar el modelo predictivo que se generó al inicio. En el apartado 5.1.1 *Generar modelo predictivo* se describen las características de este.
- 2- Después de cargar el modelo, el servidor ya puede lanzarse con el fin de esperar peticiones.
- 3- El servidor en este punto estará esperando peticiones de algún módulo cliente, ya que al estar separados puede tratar peticiones de distintos dispositivos.
- 4- Cuando el servidor recibe una petición comprueba en primer momento si esta contiene toda la información necesaria: la imagen del rostro y el área de la imagen en la que se encuentra el rostro. Si contiene esta información se continuaría con la ejecución. Si no es así, responde a la petición con un mensaje de “error” y vuelve al paso 3 a esperar más peticiones.
- 5- En este momento el servidor ha recibido una petición que contiene una imagen de un rostro y la situación de este. La tarea que se lleva a cabo es obtener toda la información de la imagen del rostro de forma similar a la que se determinó en el apartado 5.1.1 *Generar modelo predictivo*.
- 6- Una obtenida toda la información, se puede predecir el estado de ánimo de ese rostro con el modelo predictivo.

- 7- Por último, se envía el resultado de la predicción al cliente, se genera un log en el que se registran las operaciones llevadas a cabo durante el manejo de la petición; y se vuelve al paso 3 a esperar nuevas peticiones.

5.1.5 Comunicación entre Cliente y Servidor

Como se ha mencionado anteriormente, el módulo servidor se ha creado como una aplicación web. De esta forma, si el servidor se encontrase en un dispositivo externo, múltiples módulos clientes podrían acceder desde distintos dispositivos para consumir sus recursos si sus aplicaciones dispusieran de conexión a internet. De no disponer de conectividad, se usaría un módulo servidor alojado en el propio vehículo, de forma que se encontrarían en la misma red. En ambos casos, el intercambio de información se lleva a cabo de la misma forma:

El módulo cliente debe identificar un rostro en la imagen. Cuando identifica uno, este lo recorta y lo prepara para enviárselo al servidor. El proceso para llevar a cabo esto ya se explicó en el apartado 5.1.3 *Cliente*. Por tanto, la información que debe enviar el módulo cliente al módulo servidor es la imagen recortada con el rostro del usuario identificado. Esta imagen, para poder ser transmitida se introduce en un *json*, que es un formato de texto ligero creado para el intercambio de datos entre aplicaciones web. [24] Para que el servidor prediga el estado de ánimo de esa imagen, el cliente debe enviarle una petición “*predict*” de tipo *Post* con la información anterior.

Por otro lado, cuando el módulo servidor se ha iniciado y está ejecutándose de forma correcta, está esperando peticiones de algún módulo cliente, ya sea ejecutando en local o en línea. Como se ha mencionado anteriormente, de los módulos clientes va a recibir siempre peticiones “*predict*” con un *json* en el que se encuentra la imagen de un rostro ya recortado. Como la información va contenida en un *json*, lo primero que debe hacer para poder extraer información de la imagen es convertirla a un formato correcto, es decir, a una matriz. Tras esto, se puede extraer la información y determinar el estado de ánimo del rostro como se ha explicado anteriormente. Por último, envía al cliente el estado de ánimo resultante en un *json* de nuevo.

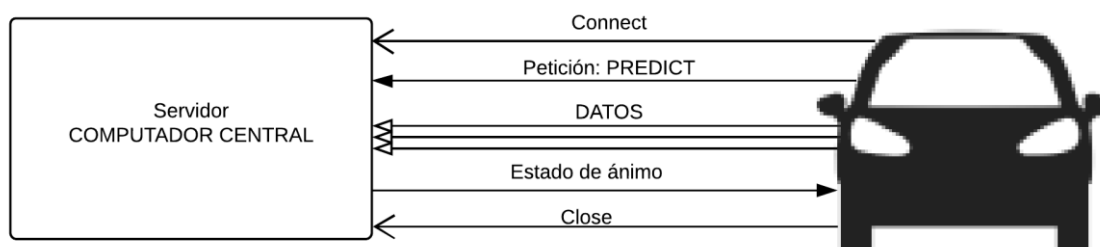


Ilustración 22. Ejemplo de conexión cliente servidor.

5.2 Despliegue

En esta sección se exponen todos los requerimientos tanto software como hardware para el despliegue de la herramienta con un funcionamiento correcto:

- Sistema operativo: para el correcto funcionamiento de la herramienta se recomienda el uso de Ubuntu 16.04 LTS.
- Procesador: los requisitos de procesador pueden variar, en función de si la herramienta va a ser ejecutada en local o se conectará a un servidor externo. Si se ejecuta todo en local se recomienda disponer de un procesador Intel® Core™ i5-6600 o superior; por otro lado, si el servidor se va a alojar en otro dispositivo externo, sería recomendable que ese dispositivo dispusiera de un procesador superior como el Intel® Core™ i7-8750, mientras que el dispositivo cliente podría bastar con un Intel® Core™ i3-6300.
- Memoria RAM: se recomienda disponer de al menos 2 GB de RAM.
- Almacenamiento: se requieren al menos 20 GB de almacenamiento. La principal causa de esto es el almacenamiento de *logs* cuando se ejecuta la herramienta en local.
- Conexión red: es aconsejable disponer de una buena conexión a la red para que el intercambio de información entre cliente y servidor sea lo más rápido y eficiente posible. Por eso se recomienda disponer de conexión 4G.

Por otro lado, para que el software sea compatible, es imprescindible tener instaladas en los dispositivos que se vaya a ejecutar las siguientes herramientas:

- Python 3.5: es el lenguaje de programación usado para el desarrollo del proyecto.
- Dlib 19: es una de las bibliotecas de *Machine Learning* empleadas para la extracción de información y la generación de conocimiento.

Scikit-learn: es la biblioteca de *Machine Learning* empleada para entrenar el modelo predictivo.

- Flask: es el *framework* de Python empleado para crear la aplicación web (cliente-servidor).
- OpenCV 3.1: es la principal biblioteca de Visión Artificial desarrollada hasta el momento; además de ser la empleada para el desarrollo de este proyecto.

6. Verificación, validación y evaluación

En este apartado se describe el proceso de verificación, validación y evaluación del proyecto. Para ello se disponen un conjunto de pruebas que permitan comprobar que el sistema cumple con el conjunto de requisitos que se detallan en el apartado 3 de la memoria: Análisis.

Para la realización y evaluación de distintas las pruebas se ha usado un ordenador doméstico con una webcam integrada. Con esto podría ser suficiente para demostrar que el sistema funciona y que, posteriormente, se podría integrar en un vehículo.

6.1 Pruebas de verificación

Con las pruebas de verificación se desea comprobar si el sistema que se está construyendo se está haciendo correctamente. Cada prueba de verificación será descrita por una tabla como la plantilla que se muestra a continuación:

IDENTIFICADOR		VET-XX	
Nombre			
Requisitos			
Descripción			
Precondiciones			
Procedimiento			
Postcondiciones			
Evaluación			

Tabla 53. Ejemplo de tabla de prueba de verificación (VET-XX).

Donde cada campo indicará lo siguiente:

- **Identificador:** Cada prueba de verificación estará identificado por las siglas VET (*Verification Test*). "XX" identificará el número de la prueba de verificación, siendo "01" el primero y "99" el último.

- **Nombre:** Nombre que describirá mínimamente el propósito del test.
- **Requisitos:** Conjunto de requisitos evaluados con este test.
- **Precondiciones:** Condiciones que se deben cumplir para poder llevar a cabo la operación.
- **Procedimiento:** Conjunto de acciones a llevar a cabo para realizar el test deseado.
- **Postcondiciones:** Estado del sistema tras realizar la operación
- **Evaluación:** será el resultado del test. En caso de ser el adecuado el campo contendrá "OK"; en caso contrario "FAIL".

Las pruebas de verificación se exponen a continuación:

IDENTIFICADOR VET-01	
Nombre	Situación de la cámara
Requisitos	RSNF-01, RSNF-02
Descripción	Comprobar que la situación de la cámara respecto a la cabeza de un conductor es la adecuada
Precondiciones	<ul style="list-style-type: none"> • Disponer de un vehículo. • Disponer de una cámara. • El sistema debe estar instalado en el dispositivo.
Procedimiento	La cámara debe situarse a una distancia mínima de 40 cm del rostro del conductor en la parte superior del vehículo
Postcondiciones	<ul style="list-style-type: none"> • El sistema podrá reconocer tanto la cámara como el rostro de un usuario.
Evaluación	OK

Tabla 54.Prueba de Verificación VET-01

IDENTIFICADOR VET-02	
Nombre	Posición del rostro
Requisitos	RNF-03
Descripción	Comprobar que el rostro detectado esta contenido entre cuatro puntos al ser enviado al servidor.
Precondiciones	<ul style="list-style-type: none"> • Debe haberse entrenado un modelo predictivo. • El servidor debe estar ejecutándose. • El servidor debe cargar el modelo predictivo • El servidor y la plataforma en la que vaya a ejecutarse la parte cliente deben tener conexión.
Procedimiento	Debe ejecutarse la parte cliente del sistema y se debe llevar a cabo al menos una iteración.
Postcondiciones	<ul style="list-style-type: none"> • El servidor recibirá una petición con el rostro de una imagen. • El servidor muestra los 4 puntos recibidos en los que se encuentra el rostro.
Evaluación	OK

Tabla 55. Prueba de Verificación VET-02

IDENTIFICADOR VET-03	
Nombre	Formato del log de ejecución
Requisitos	RNF-04
Descripción	Se comprueba si el formato del log de ejecución es “.txt”.
Precondiciones	<ul style="list-style-type: none"> • Debe haberse entrenado un modelo predictivo. • El servidor debe estar ejecutándose. • El servidor debe cargar el modelo predictivo. • El servidor y la plataforma en la que vaya a ejecutarse la parte cliente deben tener conexión. • El servidor recibe una petición por parte de un cliente.
Procedimiento	Cuando el servidor ha terminado de tratar una petición de un cliente genera un log con los datos generados tras el manejo de dicha petición. Para comprobar el formato de este log se debe ir a la carpeta que almacene estos logs.
Postcondiciones	<ul style="list-style-type: none"> • Se ha generado un nuevo log de ejecución. • El formato del log es “.txt”.
Evaluación	OK

Tabla 56. Prueba de Verificación VET-03

IDENTIFICADOR VET-04	
Nombre	Frames por segundo
Requisitos	RNF-05
Descripción	Comprar el número de frames que se analizan por segundo
Precondiciones	<ul style="list-style-type: none"> • El servidor debe haberse iniciado correctamente. • El servidor y la plataforma en la que vaya a ejecutarse la parte cliente deben tener conexión.
Procedimiento	Se debe ejecutar el cliente en modo debug para poder ver así el número de frames por segundo que se analizan.
Postcondiciones	<ul style="list-style-type: none"> • Se analizan al menos 5 frames por segundo.
Evaluación	OK

Tabla 57. Prueba de Verificación VET-04

IDENTIFICADOR VET-05	
Nombre	Comprobar el conjunto de entrenamiento
Requisitos	RNF-06
Descripción	Se comprueba que cada subconjunto de entrenamiento contenga el número muestras mínimo indicado.
Precondiciones	<ul style="list-style-type: none"> • Disponer de una base de datos de imágenes ordenadas por estado de ánimo.
Procedimiento	Antes de entrenar el modelo de predicción, hacer una comprobación del número de imágenes que se dispone para entrenar para cada estado de ánimo, y en caso de ser inferior al número descrito en el apartado de requisitos, descartar dicho subconjunto.
Postcondiciones	<ul style="list-style-type: none"> • Se descartan estados de ánimo con pocas muestras de entrenamiento y se usan los que cumplen el requisito.
Evaluación	OK

Tabla 58. Prueba de Verificación VET-05

IDENTIFICADOR VET-06	
Nombre	Formato y dimensiones de imágenes de entrenamiento.
Requisitos	RNF-07
Descripción	Se comprueba el formato y las dimensiones de las imágenes de la base de datos de las que se dispone para entrenar el modelo predictivo.
Precondiciones	<ul style="list-style-type: none"> • Disponer de una base de datos de imágenes de rostros. • Ordenar las imágenes por estados de ánimo
Procedimiento	Se debe abrir con el explorador de archivos la carpeta contenedora de las imágenes y comprobar que todas son del mismo formato y dimensiones.
Postcondiciones	<ul style="list-style-type: none"> • Se dispone de una base de datos de imágenes de rostros adecuada para entrenar un sistema.
Evaluación	OK

Tabla 59. Prueba de Verificación VET-06

IDENTIFICADOR VET-07	
Nombre	Estados de ánimo reconocidos.
Requisitos	RNF-08
Descripción	Se comprueba que se reconocen 6 estados de ánimo diferentes.
Precondiciones	<ul style="list-style-type: none"> • Debe haberse entrenado un modelo predictivo. • El servidor debe estar ejecutándose. • El servidor debe cargar el modelo predictivo • El servidor y la plataforma en la que vaya a ejecutarse la parte cliente deben tener conexión. • Tomar fotos de rostros de sujetos diferentes a los del conjunto de entrenamiento con los estados de ánimo que se desea detectar con el sistema.
Procedimiento	Ejecutar el cliente con las distintas fotos y comprobar el resultado.
Postcondiciones	<ul style="list-style-type: none"> • Se detectan los distintos estados de ánimo.
Evaluación	OK

Tabla 60. Prueba de Verificación VET-07

IDENTIFICADOR VET-08	
Nombre	Ejecución offline
Requisitos	RNF-10
Descripción	Se comprueba que el sistema funciona sin conexión a internet.
Precondiciones	<ul style="list-style-type: none"> • El servidor debe estar alojado en el mismo dispositivo que la parte cliente del sistema. • El servidor debe estar ejecutándose. • El servidor debe disponer de un modelo predictivo.
Procedimiento	Ejecutar el cliente
Postcondiciones	<ul style="list-style-type: none"> • El cliente se ejecuta correctamente. • El sistema funciona con normalidad.
Evaluación	OK

Tabla 61. Prueba de Verificación VET-08

IDENTIFICADOR VET-09	
Nombre	Ejecución con conexión a internet
Requisitos	RNF-09
Descripción	Se comprueba que el sistema funciona con conexión a internet.
Precondiciones	<ul style="list-style-type: none"> • El sistema dispone de conexión a internet. • La parte cliente del sistema es capaz de conectarse a un servidor externo. • El servidor debe estar ejecutándose. • El servidor debe disponer de un modelo predictivo.
Procedimiento	Ejecutar el cliente
Postcondiciones	<ul style="list-style-type: none"> • El cliente se ejecuta correctamente. • El sistema funciona con normalidad.
Evaluación	OK

Tabla 62. Prueba de Verificación VET-09

IDENTIFICADOR VET-10	
Nombre	Funcionamiento de la cámara
Requisitos	RSF-01, RSF-15
Descripción	Comprobar que la cámara funciona y reconoce la parte frontal de un rostro.
Precondiciones	<ul style="list-style-type: none"> • Debe haberse entrenado un modelo predictivo. • El servidor debe estar ejecutándose. • El servidor debe cargar el modelo predictivo • El servidor y la plataforma en la que vaya a ejecutarse la parte cliente deben tener conexión.
Procedimiento	Debe ejecutarse el parte cliente de la aplicación.
Postcondiciones	<ul style="list-style-type: none"> • El servidor recibirá una petición con el rostro de una imagen.
Evaluación	OK

Tabla 63. Prueba de Verificación VET-10

IDENTIFICADOR VET-11	
Nombre	Obtener información de un rostro
Requisitos	RSF-02, RSF-03, RSF-04, RSF-05, RSF-10
Descripción	Comprobar que el sistema puede obtener toda la información de interés para predecir el estado de ánimo de un usuario: Facial Landmarks, inclinación y punto medio.
Precondiciones	<ul style="list-style-type: none"> • El cliente está ejecutando correctamente. • El cliente detecta un rostro. • El servidor está esperando a recibir una petición.
Procedimiento	Se lleva a cabo una ejecución del sistema en local y se comprueba en el log de ejecución que se extraen todos los datos deseados correctamente.
Postcondiciones	<ul style="list-style-type: none"> • Se predice el estado de ánimo del usuario.
Evaluación	OK

Tabla 64. Prueba de Verificación VET-11

IDENTIFICADOR VET-12	
Nombre	Comprobar interfaz gráfica
Requisitos	RSF-06, RSF-07, RSF-17, RSF-18, RSF-16
Descripción	Comprobar que cuando se ejecuta con la interfaz gráfica activada, se muestran todos los datos deseados: rectángulo que delimita la posición de la cara, Facial Landmarks, punto medio de la cara y estado de ánimo.
Precondiciones	<ul style="list-style-type: none"> • Debe haberse entrenado un modelo predictivo. • El servidor debe estar ejecutándose. • El servidor debe cargar el modelo predictivo • El servidor y la plataforma en la que vaya a ejecutarse la parte cliente deben tener conexión.
Procedimiento	Debe ejecutarse el parte cliente de la aplicación con la interfaz gráfica activada.
Postcondiciones	<ul style="list-style-type: none"> • Se puede monitorizar en tiempo real el rostro del usuario y toda la información que se extrae de él, así como el resultado de la predicción del estado de ánimo.
Evaluación	OK

Tabla 65. Prueba de Verificación VET-12

IDENTIFICADOR VET-13	
Nombre	Detectar estado de ánimo
Requisitos	RSF-08
Descripción	Comprobar que el sistema es capaz de detectar el estado de ánimo del usuario
Precondiciones	<ul style="list-style-type: none"> • Debe haberse entrenado un modelo predictivo. • El servidor debe estar ejecutándose. • El servidor debe cargar el modelo predictivo • El servidor y la plataforma en la que vaya a ejecutarse la parte cliente deben tener conexión.
Procedimiento	Debe ejecutarse el parte cliente de la aplicación y comprobar que se muestra el estado de ánimo sin que el sistema tenga errores de ejecución.
Postcondiciones	<ul style="list-style-type: none"> • El cliente detecta un rostro y envía una petición al servidor con éste. • El cliente recibe el estado de ánimo del usuario y lo muestra
Evaluación	OK

Tabla 66. Prueba de Verificación VET-13

IDENTIFICADOR VET-14	
Nombre	Modo debug
Requisitos	RSF-09, RSF-19
Descripción	Comprobar que al iniciar en modo debug se activa la interfaz gráfica.
Precondiciones	<ul style="list-style-type: none"> • El servidor debe estar ejecutándose. • El servidor debe cargar el modelo predictivo • El servidor y la plataforma en la que vaya a ejecutarse la parte cliente deben tener conexión.
Procedimiento	Debe ejecutarse el parte cliente de la aplicación en modo debug.
Postcondiciones	<ul style="list-style-type: none"> • Se activa la interfaz gráfica del sistema.
Evaluación	OK

Tabla 67. Prueba de Verificación VET-14

IDENTIFICADOR VET-15	
Nombre	Nuevo modelo predictivo
Requisitos	RSF-11, RSF-12
Descripción	Comprobar que se puede generar un nuevo modelo predictivo con nuevas muestras, algoritmos diferentes o datos diferentes.
Precondiciones	<ul style="list-style-type: none"> • Disponer de una base de datos de imágenes de rostros. • La base de datos debe estar ordenada por estados de ánimo. • Se debe disponer de un número de imágenes mínimo para cada estado de ánimo que reconoce el sistema.
Procedimiento	Debe ejecutarse el módulo de entrenamiento con la nueva configuración, entrenará un nuevo modelo y mostrará su porcentaje de acierto en el resultado.
Postcondiciones	<ul style="list-style-type: none"> • El módulo de entrenamiento guarda este nuevo modelo en memoria. • Si se va a utilizar este nuevo modelo debe actualizarse el módulo servidor, de forma que cargue este nuevo modelo. • Si el nuevo modelo emplea información diferente a la que se utilizaba en el momento, debe actualizarse la información que extrae el servidor de cada rostro, de forma que coincidan.
Evaluación	OK

Tabla 68. Prueba de Verificación VET-15

IDENTIFICADOR VET-16	
Nombre	Reporte de errores
Requisitos	RSF-13, RSF-14
Descripción	Comprobar que si se produce un error en tiempo de ejecución este se reporta de forma correcta.
Precondiciones	<ul style="list-style-type: none"> • Ejecutar el módulo cliente.
Procedimiento	Comprobar que el sistema reporta excepciones en los siguientes casos: <ul style="list-style-type: none"> • Si la cámara no está conectada o no es reconocida. • Si no es capaz de establecerse conexión con el servidor. • Si hay errores en el intercambio de información entre cliente y servidor. • Si hay algún error de ejecución diferente.
Postcondiciones	<ul style="list-style-type: none"> • Los errores aparecidos en tiempo de ejecución quedarán registrados en el log de ejecución.
Evaluación	OK

Tabla 69. Prueba de Verificación VET-16

A continuación, se muestra la matriz de trazabilidad de prueba de verificación y requisitos software:

	VET-01	VET-02	VET-03	VET-04	VET-05	VET-06	VET-07	VET-08	VET-09	VET-10	VET-11	VET-12	VET-13	VET-14	VET-15	VET-16
RSF-01	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-	-
RSF-02	-	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-
RSF-03	-	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-
RSF-04	-	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-
RSF-05	-	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-
RSF-06	-	-	-	-	-	-	-	-	-	-	-	X	-	-	-	-
RSF-07	-	-	-	-	-	-	-	-	-	-	-	X	-	-	-	-
RSF-08	-	-	-	-	-	-	-	-	-	-	-	-	X	-	-	-
RSF-09	-	-	-	-	-	-	-	-	-	-	-	-	-	X	-	-
RSF-10	-	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-
RSF-11	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	-
RSF-12	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	-
RSF-13	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X
RSF-14	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X
RSF-15	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-	-
RSF-16	-	-	-	-	-	-	-	-	-	-	-	X	-	-	-	-
RSF-17	-	-	-	-	-	-	-	-	-	-	-	X	-	-	-	-
RSF-18	-	-	-	-	-	-	-	-	-	-	-	X	-	-	-	-
RSF-19	-	-	-	-	-	-	-	-	-	-	-	-	-	X	-	-
RSNF-01	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
RSNF-02	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
RSNF-03	-	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-
RSNF-04	-	-	X	-	-	-	-	-	-	-	-	-	-	-	-	-
RSNF-05	-	-	-	X	-	-	-	-	-	-	-	-	-	-	-	-
RSNF-06	-	-	-	-	X	-	-	-	-	-	-	-	-	-	-	-
RSNF-07	-	-	-	-	-	X	-	-	-	-	-	-	-	-	-	-
RSNF-08	-	-	-	-	-	-	X	-	-	-	-	-	-	-	-	-
RSNF-09	-	-	-	-	-	-	-	-	X	-	-	-	-	-	-	-
RSNF-10	-	-	-	-	-	-	-	X	-	-	-	-	-	-	-	-

Tabla 70. Matriz de trazabilidad de pruebas VET y Requisitos Software.

6.2 Pruebas de validación

Con las pruebas de validación se desea comprobar si el sistema que se ha construido es el correcto. Cada prueba de validación será descrita por una tabla como la plantilla que se muestra a continuación:

IDENTIFICADOR	VAT-XX
Nombre	
Requisitos	
Descripción	
Precondiciones	
Procedimiento	
Postcondiciones	
Evaluación	

Tabla 71. Ejemplo de tabla de prueba de validación (VAT-XX).

Donde cada campo indicará lo siguiente:

- **Identificador:** Cada prueba de validación estará identificado por las siglas VAT (*Validation Test*). "XX" identificará el número de la prueba de validación, siendo "01" el primero y "99" el último.
- **Nombre:** Nombre que describirá mínimamente el propósito del test.
- **Requisitos:** Conjunto de requisitos evaluados con este test.
- **Precondiciones:** Condiciones que se deben cumplir para poder llevar a cabo la operación.
- **Procedimiento:** Conjunto de acciones a llevar a cabo para realizar el test deseado.
- **Postcondiciones:** Estado del sistema tras realizar la operación

- **Evaluación:** será el resultado del test. En caso de ser el adecuado el campo contendrá “OK”; en caso contrario “FAIL”.

Las pruebas de validación se exponen a continuación:

IDENTIFICADOR VAT-01	
Nombre	Identificación de estado de ánimo
Requisitos	RUC-01, RUC-02, RUC-03
Descripción	Comprobar que el sistema es capaz de reconocer el estado de ánimo de un usuario al volante.
Precondiciones	<ul style="list-style-type: none"> • Debe haberse entrenado un modelo predictivo. • El servidor debe estar ejecutándose. • El servidor debe cargar el modelo predictivo • El servidor y la plataforma en la que vaya a ejecutarse la parte cliente deben tener conexión.
Procedimiento	Ejecutar el módulo cliente y situar un rostro delante de la cámara.
Postcondiciones	<ul style="list-style-type: none"> • El cliente detecta un rostro. • El servidor emplea los Facial Landmarks para extraer la información necesaria para determinar el estado de ánimo del usuario • Se puede monitorizar en tiempo real el rostro del usuario y su estado de ánimo.
Evaluación	OK

Tabla 72. Prueba de Validación VAT-01

IDENTIFICADOR VAT-02	
Nombre	Modo debug
Requisitos	RUC-04, RUC-05, RUC-06, RUNC-06
Descripción	Comprobar que el sistema se puede ejecutar en modo debug y este muestra la información deseada.
Precondiciones	<ul style="list-style-type: none"> • El servidor debe estar en funcionamiento • El cliente en el que se vaya a ejecutar y el servidor deben tener conexión.
Procedimiento	Ejecutar el cliente en modo debug.
Postcondiciones	<ul style="list-style-type: none"> • Se activa la interfaz gráfica del sistema que permite monitorizar en todo momento el rostro situado delante de la cámara. • En caso de detectarse un rostro se muestra un rectángulo en el área en el que se reconoció, los <i>Facial Landmarks</i>, el punto medio de la cara y el estado de ánimo reconocido. • Se ve el número de peticiones por segundo que recibe el servidor para determinar el estado de ánimo del usuario.
Evaluación	OK

Tabla 73. Prueba de Validación VAT-02

IDENTIFICADOR VAT-03	
Nombre	Monitorización de errores de ejecución
Requisitos	RUC-07, RUC-09, RUNC-03
Descripción	Comprobar que el sistema informa cuando se ha producido un error de ejecución y este ha quedado registrado en el log de ejecución con formato <i>.txt</i>
Precondiciones	<ul style="list-style-type: none"> • El módulo cliente debe ejecutarse.
Procedimiento	<p>Se debe exponer el sistema a posibles situaciones de error y comprobar que se muestra el mensaje de error y este queda registrado en el log de ejecución. Algunas de estas situaciones pueden ser:</p> <ul style="list-style-type: none"> • En la que no se reconozca la cámara. • En la que no se pueda comunicar con el módulo servidor • Etc.
Postcondiciones	<ul style="list-style-type: none"> • El sistema muestra un mensaje de error corta la ejecución.
Evaluación	OK

Tabla 74. Prueba de Validación VAT-03

IDENTIFICADOR VAT-04	
Nombre	Reentrenamiento del sistema
Requisitos	RUC-08, RUNC-04, RUNC-05
Descripción	Comprobar se puede generar un nuevo modelo predictivo a partir de una base de datos de imágenes adecuada.
Precondiciones	<ul style="list-style-type: none"> • Se debe disponer de una base de datos de imágenes ordenada por estados de ánimo. • Debe haber imágenes para todos los estados de ánimo del sistema.
Procedimiento	Ejecutar el módulo de entrenamiento.
Postcondiciones	<ul style="list-style-type: none"> • Se generará un nuevo archivo en el que se almacene el nuevo modelo predictivo. • Se mostrará en pantalla el porcentaje de acierto de este nuevo modelo.
Evaluación	OK

Tabla 75. Prueba de Validación VAT-04

IDENTIFICADOR VAT-05	
Nombre	Disposición de la cámara
Requisitos	RUNC-01
Descripción	Comprobar que la cámara está bien situada respecto al rostro del usuario.
Precondiciones	<ul style="list-style-type: none"> • El sistema debe estar instalado en el dispositivo en el que vaya a ejecutarse. • La cámara debe estar conectada al sistema.
Procedimiento	La cámara debe estar situada de forma que pueda monitorizar el frontal del rostro del usuario.
Postcondiciones	<ul style="list-style-type: none"> • El sistema puede monitorizar el rostro del usuario.
Evaluación	OK

Tabla 76. Prueba de Validación VAT-05

IDENTIFICADOR		VAT-06
Nombre	Conexión con el servidor	
Requisitos	RUNC-02	
Descripción	Comprobar que el sistema puede ser ejecutado tanto de forma distribuida como en local cuando el dispositivo carezca de conexión a internet.	
Precondiciones	<ul style="list-style-type: none"> El sistema debe estar instalado en un dispositivo, tanto la parte cliente como la parte del servidor. El servidor debe estar instalado también en un dispositivo externo con conexión a internet. Ambos dispositivos deben disponer de un modelo predictivo. 	
Procedimiento	Ejecutar el módulo de entrenamiento con el dispositivo conectado a internet y sin conexión y ver como el sistema funciona correctamente.	
Postcondiciones	<ul style="list-style-type: none"> Se monitoriza el estado de ánimo del usuario en tiempo real. 	
Evaluación	OK	

Tabla 77. Prueba de Validación VAT-06

A continuación, se expone la matriz de trazabilidad de pruebas de validación y requisitos de usuario:

	VAT-01	VAT-02	VAT-03	VAT-04	VAT-05	VAT-06
RUC-01	X	-	-	-	-	-
RUC-02	X	-	-	-	-	-
RUC-03	X	-	-	-	-	-
RUC-04	-	X	-	-	-	-
RUC-05	-	X	-	-	-	-
RUC-06	-	X	-	-	-	-
RUC-07	-	-	X	-	-	-
RUC-08	-	-	-	X	-	-
RUC-09	-	-	X	-	-	-
RUNC-01	-	-	-	-	X	-
RUNC-02	-	-	-	-	-	X
RUNC-03	-	-	X	-	-	-
RUNC-04	-	-	-	X	-	-
RUNC-05	-	-	-	X	-	-
RUNC-06	-	X	-	-	-	-

Tabla 78. Matriz de trazabilidad de pruebas VAT y Requisitos de Usuario.

7. Planificación y presupuesto

En este apartado se explicará la planificación de la realización del proyecto software, justificando la metodología de desarrollo tomada; así como el presupuesto para llevar a cabo el proyecto, haciendo una estimación de costes; y, por último, se explicará el impacto socioeconómico que este puede tener.

7.1 Planificación

A continuación, se describe la planificación llevada a cabo para la realización del proyecto. En primer lugar, se justifica la metodología de desarrollo software seguida; seguido de una estimación de tiempos para el desarrollo del proyecto.

7.1.1 Justificación de la metodología

“La metodología de desarrollo de software es un marco de trabajo usado para estructurar, planificar y controlar el proceso de desarrollo en sistemas de información [21]”. Es por esto que es importante fijar una metodología que ayude a enfocar el desarrollo del proyecto.

Se han creado multitud de metodologías diferentes que se adaptan a las necesidades de distintos tipos de proyectos, objetivos y clientes en función de las necesidades de cada uno; hay desde metodologías estructuradas, hasta metodologías orientadas a sistemas de tiempo real, o metodologías ágiles.

En esta situación concreta, para este proyecto, al tener los objetivos claros desde un principio sobre las intenciones del proyecto, se ha optado por una metodología estructurada como es la metodología en cascada.

El desarrollo en cascada es un modelo de desarrollo secuencial con distintas etapas; de tal forma que el inicio de cada etapa debe esperar a la finalización de la etapa anterior. Además, Al final de cada etapa se lleva a cabo una revisión final con el fin de determinar si el proyecto está listo para avanzar a la siguiente fase. [22]

La siguiente imagen muestra el modelo de cascada elegido, con todas las fases con las que se ha contado para el desarrollo del proyecto:

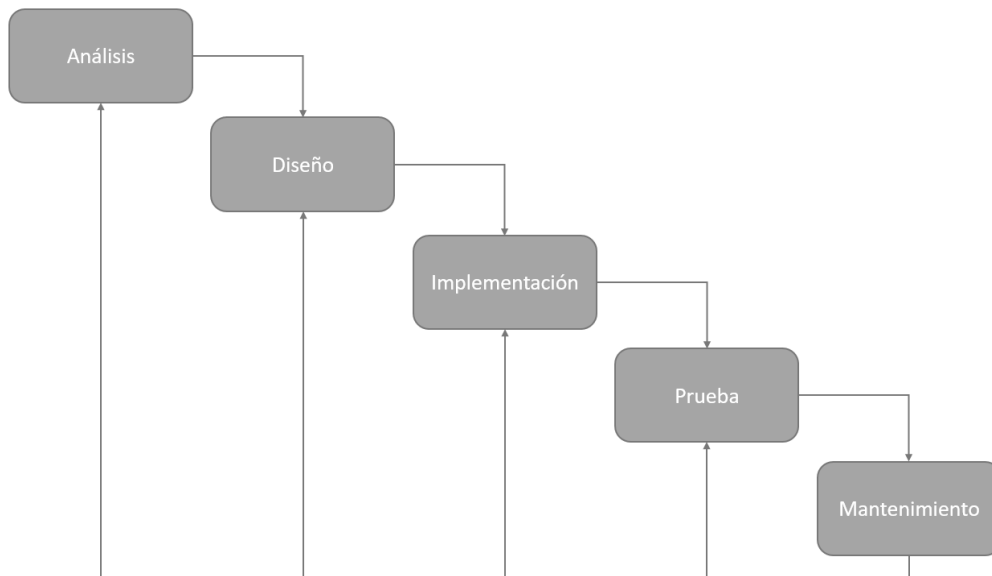


Ilustración 23. Diagrama de desarrollo en cascada.

A continuación, se detallará individualmente cada una de las etapas con las que cuenta el modelo de desarrollo en cascada elegido:

- **Análisis:** en esta fase se analizan las necesidades de los usuarios finales del sistema para determinar qué objetivos debe cubrir. De esta fase surgen los requisitos, que contienen la información de lo que debe realizar el sistema.
- **Diseño:** en esta etapa se describe la estructura interna del software y las relaciones entre los distintos módulos que lo componen. Descompone y organiza el sistema en módulos que pueden ser elaborados por separado.
- **Implementación:** En esta fase se programan los requisitos especificados haciendo uso de las estructuras de datos diseñadas en la fase anterior.
- **Prueba:** en esta fase se comprueba que los componentes del sistema implementados funcionan correctamente y cumplen con los requisitos del sistema.

- **Mantenimiento:** En esta fase se comprueba que el sistema funciona correctamente en el entorno que se vaya a utilizar. A partir de aquí, se deben destinar recursos a mantener el software, ya sea corrigiendo errores o para mejorar el rendimiento y las características.

7.1.2 Tiempo estimado

A continuación, se expone el tiempo tomado para realizar cada una de las fases anteriormente descritas detallando, además, las tareas realizadas en cada una de ellas.

ETAPA	DURACIÓN (HORAS)	FECHA DE INICIO	FECHA DE FIN
ANÁLISIS	20	02/06/2017	28/06/2017
REUNIÓN CON TUTOR	5	02/06/2017	19/06/2017
ESPECIFICACIÓN DE REQUISITOS	15	20/06/2017	28/06/2017
DISEÑO DEL SISTEMA	50	03/07/2017	01/08/2017
ARQUITECTURA DEL SISTEMA	50	03/07/2017	01/08/2017
IMPLEMENTACIÓN DEL SISTEMA	140	26/08/2017	09/10/2017
MÓDULO DE ENTRENAMIENTO	70	26/08/2017	12/09/2017
MÓDULO SERVIDOR	20	14/09/2017	20/09/2017
MÓDULO CLIENTE	30	21/09/2017	30/09/2017
COMUNICACIÓN ENTRE MÓDULOS	20	01/10/2017	09/10/2017
PRUEBAS	30	04/11/2017	30/11/2017
VERIFICACIÓN DEL SISTEMA	17	04/11/2017	20/11/2017
VALIDACIÓN DEL SISTEMA	13	21/11/2017	30/11/2017
DOCUMENTACIÓN	60	18/01/2018	30/05/2018
TOTAL	300	02/06/2017	30/05/2018

Tabla 79. Tabla de estimación de tiempos.

Destacar que la fecha de comienzo de la realización del proyecto fue el 02/06/2017 y la de finalización el 30/04/2018. El tiempo y los días destinados a la realización han sido variables; y ha habido intervalos de tiempo en los que, por causas externas se ha suspendido la realización del proyecto.

A continuación, se expone el diagrama de Gantt con los mismos datos, pero que ayudará a comprender de manera más visual los datos anteriores.

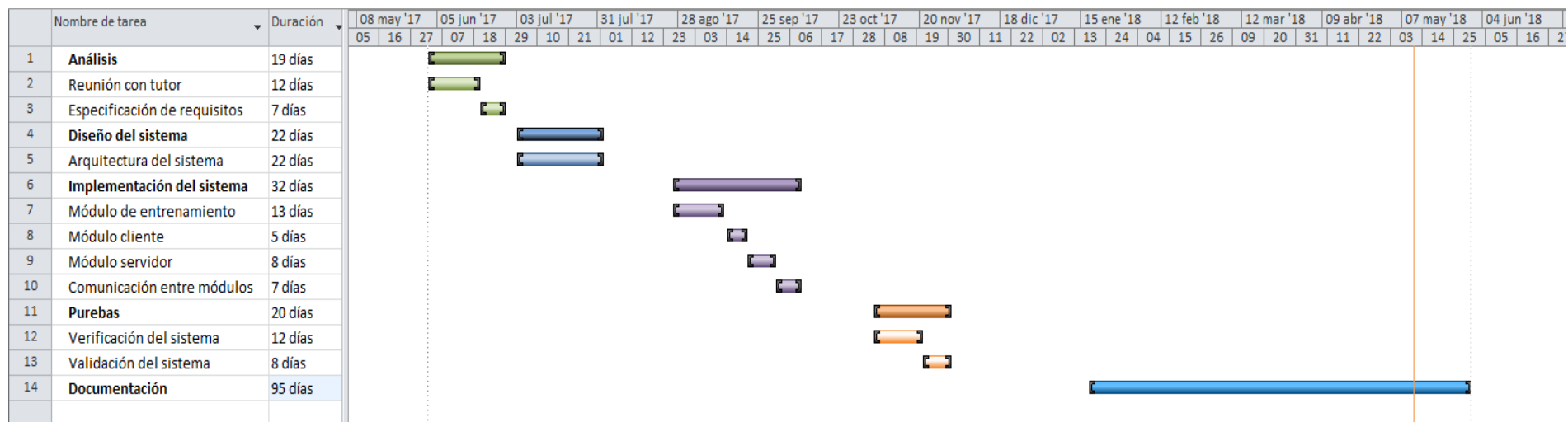


Ilustración 24. Diagrama de Gantt de la planificación del proyecto.

7.2 Presupuesto

En este apartado se exponen costes de llevar a cabo este proyecto. Se comenzará analizando los costes parciales del proyecto, que se dividirán en costes materiales y costes personales; y por último se expondrán los costes globales.

7.2.1 Costes parciales

Los costes parciales se dividen en costes materiales y costes personales. Ambos quedan expuestos a continuación:

7.2.1.1. Costes Materiales

Los costes materiales vienen dados por el precio de los dispositivos utilizados tanto para su desarrollo, como para realizar las pruebas. Para ello es importante conocer el tiempo de uso de los distintos componentes, así como su precio individual y tiempo de amortización.

El coste real que tendrá en el proyecto vendrá dado por el precio de cada dispositivo, dividido por su tiempo de amortización y multiplicado por el tiempo de uso.

Los costes materiales se muestran en la siguiente tabla:

Dispositivo	Precio (€)	Tiempo de amortización (meses)	Tiempo de uso (meses)	Coste (€)
PC de desarrollo	1600,00	24	10	666,67
PC de simulación	659,00	24	5	137,30
KINECT	50,00	24	5	10,42
TOTAL				814,39

Tabla 80. Tabla de costes materiales.

Como se puede comprobar en la tabla, los costes materiales suman un total de 814,39 €.

7.2.1.2. Costes personales

Por otro lado, además de los costes del material empleado, hay que tener en cuenta el sueldo de las personas que han contribuido en el proyecto. Además, se debe tener en cuenta el rol de cada uno en las distintas etapas para fijar sueldos. El tutor del proyecto y el estudiante han desempeñado los siguientes roles:

- Tutor: jefe de proyecto.
- Estudiante: analista, diseñador, desarrollador y *tester*.

A continuación, se muestran los sueldos de cada uno:

CATEGORÍA	COSTE POR HORA (€)	TIEMPO (H)	TOTAL (€)
JEFE DE PROYECTO	50	25	1250,00
ANALISTA	28	30	840,00
DISEÑADOR	30	60	1800,00
DESARROLLADOR	22	155	3410,00
TESTER	25	40	1000,00
TOTAL			8300,00

Tabla 81. Tabla de costes de personal.

Como se puede comprobar en la tabla, los costes de personal suman un total de 8300,00 €.

7.2.2 Costes totales

El total de costes directos sería la suma de costes materiales y costes de personal; estos suman la cifra de 9114,34 €.

El coste total se calcula a partir de los costes materiales y personales; pero también deben tenerse en cuenta los costes indirectos, un margen de riesgo, un margen de beneficio e impuestos. Los porcentajes de costes estimados para estos son los siguientes:

- Costes indirectos: 15 %. Costes de luz, internet, transporte, etc.
- Margen de riesgo: 20%. Para cubrir imprevistos.
- Beneficios: 20 %. Los beneficios generados por el proyecto.
- IVA: 21%.

En la siguiente tabla se muestran los costes totales del proyecto:

CONCEPTO	CANTIDAD (€)
COSTES DIRECTOS	9114,34
COSTES INDIRECTOS (15%)	1367,16
MARGEN DE RIESGO (20%)	1822,87
BENEFICIOS (20%)	1822,87
BASE IMPONIBLE	14127,24
IVA 21%	2966,72
COSTE TOTAL DEL PROYECTO	17093,96

Tabla 82. Tabla de Coste Total del proyecto.

Como se puede comprobar en la tabla, el coste total del proyecto suma un total de 17.093,96 €.

7.3 Impacto socioeconómico

En este apartado se expone el posible impacto del proyecto en distintos ámbitos, tanto a nivel social como económico. Para ello, el primer aspecto a tener en cuenta sería la implantación del proyecto en los vehículos de nueva generación: en este punto ya se generarían nuevos puestos de trabajo, especializados tanto en el uso, como en el mantenimiento y el montaje de la herramienta. Por otro lado, el hecho de que la herramienta este integrada en ciertos automóviles podría motivar también a los compradores de vehículos nuevos a decantarse por uno de estos modelos, buscando beneficios tanto en la experiencia de la conducción como en seguridad; dando lugar, a su vez, beneficios a las marcas que integrasen la tecnología.

Como se acaba de mencionar, el proyecto puede tener un gran impacto en la experiencia de la conducción, pues podría adaptarse no solo al conductor en un momento dado; si no en todo momento en función de cómo se encuentre este. De esta forma el conductor tendría una experiencia totalmente adaptada personalizada cada vez que arranca su vehículo.

Teniendo en cuenta el punto anterior, se puede llegar a que la experiencia de la conducción sería más placentera. Esto quiere decir que se podrían reducir en gran medida las situaciones tanto de estrés como de conflicto durante el tiempo de conducción, dando lugar a una conducción más segura.

Reduciendo estrés y conflictos y ayudando al conductor a mantenerse concentrado en la carretera podrían llegar a reducirse notablemente la cantidad de accidentes de tráfico y todo lo que esto puede conllevar.

En primer lugar, se podría reducir el coste humano, tanto en muertes, como en heridos por accidentes. Y no solo para los propios ocupantes de los vehículos, sino también para los peatones. Además, disminuirían las afectaciones familiares que conllevan esta clase de accidentes.

Por otro lado, habría que tener en cuenta los gastos públicos que tienen lugar en cada accidente: tanto en policía, bomberos, personal sanitario, reparación de la vía, etc. Estos, podrían verse reducidos, permitiendo, por ejemplo, ser invertidos en mejorar los servicios públicos.

Reduciendo la cantidad de accidentes también disminuirían tanto las ventas de nuevos vehículos que se producen tras un siniestro, como las reparaciones en talleres.

Y, por último, mencionar el impacto medioambiental que podría tener la reducción de accidentes de tráfico. Pues ya no solo el daño que puede ser producido por un siniestro, sino por la cantidad de residuos generados y gases que se liberan.

8. Conclusiones y trabajos futuros

En este último apartado se exponen las conclusiones finales del proyecto tras su diseño, desarrollo e implementación; además de comentar unas líneas de trabajo de futuras hacia las que podría orientarse el proyecto.

8.1 Conclusiones

En el trabajo se ha descrito el desarrollo de un sistema avanzado de asistencia a la conducción capaz de reconocer el estado de ánimo de un conductor. En este momento es necesario comprobar si se han completado los objetivos esperados del proyecto y determinar si, en efecto, el sistema es capaz de reconocer el estado de ánimo de un conductor. Para ello, se debe comenzar desde las tareas realizadas más básicas hasta, en su última etapa, revisar si el objetivo final se ha cumplido.

Para comenzar, se ha conseguido entrenar un sistema capaz de determinar el estado ánimo de una persona a través su rostro con más de 500 imágenes para un total de 6 emociones. Para ello se han realizado distintas pruebas y se ha obtenido tanto el algoritmo que mayor acierto daba como la información más relevante para determinar el estado de ánimo a través de imágenes del rostro de una persona. Esta ha sido la parte más costosa de realizar, además de ser la que más trabajo de investigación ha conllevado; pues generar un buen modelo predictivo era fundamental para todo el trabajo posterior. Además, destacar que encontrar una base de datos de imágenes adecuadas para esto ha sido costoso.

A partir de aquí, generar el resto del sistema ha requerido trabajo, pero la base del proyecto ya era sólida. Entonces conectar una parte que controla una cámara y detecta rostros con otra capaz de obtener información de dicho rostro y determinar su estado de ánimo eran los siguientes pasos requeridos. Para ello se empleó *Flask*, de *Python*, permitiendo ejecutar las distintas partes del sistema en dispositivos diferentes.

Ejecutar las distintas partes del sistema en dispositivos diferentes era una tarea importante para lograr un funcionamiento eficiente del sistema, sin sobrecargar el del propio vehículo. Esta tarea se ha logrado llevar a cabo con un flujo de imágenes correcto. Y, destacar, que el resultado es igualmente satisfactorio cuando las distintas partes del sistema se ejecutan en el mismo

dispositivo, aunque dicho flujo seguramente se vea algo reducido cuando se ejecute en dispositivos con menos recursos que los empleados para la realización del proyecto.

Para finalizar a nivel técnico, destacar que se ha cumplido el objetivo principal del sistema y se detecta de forma bastante precisa el estado de ánimo de un conductor, permitiendo ahora al vehículo controlar esta información. De esta forma, en un futuro, el vehículo podrá tomar acciones a partir de esto.

En el aspecto personal, se ha tomado conciencia de la dificultad y del trabajo que conlleva realizar un proyecto de este tipo. A pesar de haber tenido contacto anterior con el desarrollo de aplicaciones de en el que el uso de *visión artificial* y *machine learning* fuera necesario, el hecho de que en esta ocasión se tratara de monitorizar el rostro humano y, a partir de este, obtener su emoción ha tenido un punto de dificultad añadida. A pesar de esto, el resultado ha sido satisfactorio y la realización del proyecto ha sido un reto muy interesante.

Por último, mencionar que los distintos conocimientos adquiridos durante el grado en ingeniería informática han sido útiles para el desarrollo del proyecto y que, sin el tiempo de prácticas en empresa y el conocimiento adquirido en el departamento de *Everis Spain* destinado al desarrollo de proyectos de inteligencia artificial, la realización del proyecto hubiera sido notablemente más costosa.

8.2 Trabajos futuros

El sistema avanzado de asistencia a la conducción que se ha desarrollado es un primera y pequeña aproximación a la monitorización del usuario de un vehículo. Hay distintos aspectos en los que se podría profundizar más y seguir desarrollando:

- En primer lugar, se podría identificar al conductor del vehículo y monitorizar también su nivel de distracción. Que el sistema sea capaz de determinar si el usuario está dormido o si tiene su atención puesta en la conducción.
- Lo siguiente podría ser monitorizar tanto el comportamiento como el estado de ánimo de los demás ocupantes del vehículo.
- Otro elemento a tener en cuenta en el desarrollo sería conseguir que el sistema fuera capaz de determinar el estado de ánimo de los usuarios también en situaciones lumínicas adversas, como por ejemplo durante la noche.
- Obtener una base de datos con mayor cantidad de muestras de rostros para las distintas emociones detectadas podría aportar también una mejora en la precisión a la hora de detectar estados de ánimo.
- Conseguir que los cálculos más complejos que se realizan con las imágenes fueran llevados por la tarjeta gráfica en lugar del procesador optimizaría en gran medida el rendimiento del sistema en general.
- Por último, que el sistema pudiera llegar a tener control sobre distintos elementos del vehículo, ya fuera la radio, la posición de los asientos, retrovisores, incluso en algún momento el freno podría ser un aspecto muy atractivo en vehículos en los que ya se ha integrado el sistema.

9. Bibliografía

- [1] OMS, *Informe de siniestralidad vial*, 2015
- [2] MORENO F., *El empleo de cinturones de seguridad será obligatorio en todo momento, incluso en los asientos traseros*, *El País*, 1990.
- [3] DEKRA AUTOMOVIL GMBH, *Informe sobre la seguridad vial 2016*, 2016.
- [4] PAUL A., ET AL. *Advanced driver assistance systems*. *SAE Technical Paper*, 2016.
- [5] SEGURIDAD VIAL, *El Sistema ADAS ayuda a prevenir accidentes de tráfico*, 2017. URL: <https://www.seguridad-vial.net/vehiculo/seguridad-pasiva/156-el-sistema-adas-ayuda-a-prevenir-accidentes-de-trafico-a-los-conductores> (Último acceso: 28/05/2018)
- [6] DGT, *Informe y análisis sobre influencia de los sistemas de ayuda a la conducción en la seguridad vial y su aplicación para la clasificación de vehículos*, 2016. URL: <http://revista.dgt.es/images/INFORME-SISTEMAS-DE-AYUDA-A-LA-CONDUCCION-copia.pdf> (Último acceso: 28/05/2018)
- [7] DHALL A, et al. *Video and image based emotion recognition challenges in the wild: Emotiiv 2015*. En *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*. ACM, 2015. p. 423-426.
- [8] DOERRFELD B., *20+ Emotion Recognition APIs That Will Leave You Impressed, and Concerned*, 2016. URL: <https://nordicapis.com/20-emotion-recognition-apis-that-will-leave-you-impressed-and-concerned/> (Último acceso: 28/05/2018)
- [9] BISMART, *Soluciones business intelligence*. URL: <https://bismart.com/es/soluciones-business-intelligence/> (Último acceso: 28/05/2018)
- [10] BISMART, URL: <https://bismart.com/es/> (Último acceso: 28/05/2018)
- [11] AFFECTIVA, *Products*. URL: <https://www.affectiva.com/what/products/> (Último acceso: 28/05/2018)
- [12] MCDUFF D., ET AL. *AFFDEX SDK: a cross-platform real-time multi-face expression recognition toolkit*. En *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*. ACM, 2016. p. 3723-3726.

- [13] AFFECTIVA, Emotion AI Overview. URL: <https://www.affectiva.com/emotion-ai-overview/> (Último acceso: 28/05/2018)
- [14] WIKI INGENIERIA DE SOFTWARE, *¿Cuál es la diferencia entre los Requerimientos del Usuario y los Requerimientos del Sistema? plantear ejemplos.* URL: [http://es.isrequerimientosut.wikia.com/wiki/1.%C2%BFCu%C3%A1l es la diferencia entre lo s Requerimientos del Usuario y los Requerimientos del Sistema%3F plantear ejemplos.](http://es.isrequerimientosut.wikia.com/wiki/1.%C2%BFCu%C3%A1l_es_la_diferencia_entre_lo_s_Requerimientos_del_Usuario_y_los_Requerimientos_del_Sistema%3F_plantear_ejemplos) (Último acceso: 29/05/2018)
- [15] BOE, *Ley Orgánica 15/1999, de 13 de diciembre, de Protección de Datos de Carácter Personal*, 1999.
- [16] DGT, *Instruccion 15/V-113 - Autorización de pruebas o ensayos de investigación realizados con vehículos de conducción automatizada en vías abiertas al tráfico en general*, 2015.
- [17] OPENCV, *Open Source Computer Vision doc*, 2017. URL: <https://docs.opencv.org/3.4.0/index.html> (Último acceso: 29/05/2018)
- [18] OPENCV, *Face landmark detection in an image*, URL: https://docs.opencv.org/3.4.0/d2/d42/tutorial_face_landmark_detection_in_an_image.html (Último acceso: 29/05/2018)
- [19] LUCEY, P., COHN, J. F., KANADE, T., SARAGIH, J., AMBADAR, Z., & MATTHEWS, I. *The Extended Cohn-Kanade Dataset (CK+): A complete expression dataset for action unit and emotion-specified expression*. Proceedings of the Third International Workshop on CVPR for Human Communicative Behavior Analysis, 2010, San Francisco, USA, 94-101.
- [20] SRIVASTAVA N, ET AL. *Dropout: A simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research*, 2014, vol. 15, no 1, p. 1929-1958. [21] GARCÍA, E, ET AL. *Metodología para el desarrollo de software multimedia educativo MEDESME. CPU-e. Revista de Investigación Educativa*, 2016, no 23, p. 216-226.
- [22] AGUILERA, S. *Ingeniería del Software-IS*. 2015.
- [23] LORD D., MÖNNICH AD, RONACHER A., UNTERWADITZER M., *Flask*, URL: <http://flask.pocoo.org/> (Último acceso: 29/05/2018)
- [24] JSON.ORG, *Introducing JSON*, URL: <https://www.json.org/> (Último acceso: 29/05/2018)
- [25] PYTHON, *Python 3 Documentation contents*, URL: <https://docs.python.org/3/contents.html> (Último acceso: 29/05/2018)

- [26] LEO, MARCO, ET AL. *Computer vision for assistive technologies. Computer Vision and Image Understanding*, 2017, vol. 154, p. 1-15.
- [27] DHALL, ABHINAV, ET AL. *From individual to group-level emotion recognition: EmotiW 5.0. En Proceedings of the 19th ACM International Conference on Multimodal Interaction*. ACM, 2017. p. 524-528.
- [28] KULKARNI SS, REDDY NP, HARIHARAN S. *Facial expression (mood) recognition from facial images using committee neural networks*. BioMedical Engineering OnLine. 2009; 8:16. doi:10.1186/1475-925X-8-16.
- [29] Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011. URL: http://scikit-learn.org/stable/supervised_learning.html#supervised-learning (Último acceso: 29/05/2018)
- [30] DLIB, URL: <http://dlib.net/> (Último acceso: 29/05/2018)

Abstract

Introduction

Nowadays, traffic accidents are still one of the main causes of death. In the world, around 1,250,000 people die because of these kind of accidents, but it can be highlighted that the mortality rate of these accidents depends a lot on the geographic region they happened.



Illustration XXV. Deceased by traffic accident depending on the geographic region (Source: www.revista.dgt.es).

How it can be observed on *Illustration I*, the probability of dying because of a traffic accident is much higher in geographic regions like Africa or Eastern Mediterranean than in Europe, for example. This difference can be due, mainly, to the average age of the automobiles in each region, the safety systems that these vehicles include and the lack of regulation by many governments in the countries of these regions. Europe, compared to most of the regions that are showed in the map, is a more developed area due to its greater wealth. The rest of the regions, include a large number of underdeveloped countries or those that are still developing, being able to concentrate the wealth in certain points, but not being commonplace. This allows that, in richer areas, the average age of cars as in Europe, is much lower than in Africa. In

addition, bearing this in mind, the widespread incorporation of basic safety elements, such as seatbelts, was not obligatory in Spain until 1990; This means that most vehicles of previous manufacture did not even have this safety element.

In most of these underdeveloped regions, the use some security basics that are elementary for most of the developed countries culture, are not mandatory yet. And given the age of their vehicles, many also lack of these basic security mechanisms.

On the other hand, 49% of fatal victims of road traffic crashes are pedestrians, cyclists and motorcyclists, which do not have as advanced security mechanisms as those available for cars; most of them, do not even have any security mechanism. In addition, it should also be mentioned that pedestrians correspond to 22% of all victims that take place in these accidents, without even taking place inside a vehicle.

It is worthy to be mentioned that the human factor is still the origin of 90% of traffic accidents: either by distraction, emotional factor, driving under the influence of alcohol or drugs, speeding, etc. This suggests that a large part of these traffic crashes could be avoided with the use of safety systems that helped the driver to maintain attention while driving, in order to reduce those accidents that may occur due to driver distraction; and with a more rigorous regulation for those crashes that could potentially occur due to non-compliance with traffic laws and regulation.

This way, investigating and developing new systems that provide security to the vehicles of current manufacture, which also help vehicle drivers to reduce the situations of distraction and increase their attention in driving, not only reduce the number of accidents, but also the number of deaths in these. In order to accomplish this aim, it is also important to improve the regulation of the use of safety elements in vehicles and that their incorporation is economically sustainable and accessible to everybody.

Objectives

The main objective of this project is the development of an Advanced System for Driver Assistance, which, predicting the emotional state of the driver, helps to reduce situations of distraction when driving a vehicle.

This system will track the driver's face using a camera, being able to determine the mood of the driver. In this way, the vehicle will have information about the emotional state of the driver and could determine the predisposition of this one to be distracted from this activity; being able, in the future, to take actions inside the vehicle to reduce this bad predisposition.

On the other hand, its noteworthy that the system not only allows to generate a more serene driving with fewer distractions, it can also provide an improvement in the experience of driving a vehicle, as it could deliver a high degree of personalization and adaptation for each user inside a vehicle.

Architecture

The system is composed of three modules. Each of them performs a main function for the system and the communication among them is always carried out by the server module, which will be one of these modules. The other two modules will be the training module and the client module. The following diagram shows these modules and their relationships:

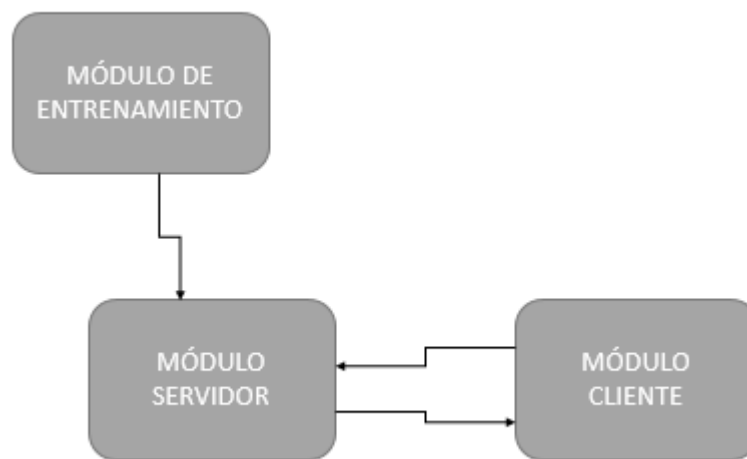


Illustration XXVI. System architecture diagram.

- The **training module** contains the algorithm that will be used to extract information from the images of the users faces. Before that, it is required the system has access to a database of faces classified by states of mind. Once the algorithm and the database are ready, the training module is able to generate a predictive model that, later, will be used by the server module to determine the mood of the users.
- The **client module** will be the main responsible for all the camera operations. It starts the camera and extracts a continuous stream of images from it. If it detects a face in the image, it cuts out the area of it, sends it to the server module and waits for its response. The response will contain the mood of the face that was detected at the beginning. This

module also controls possible execution errors that may occur with the camera, both at run time, as in case of not being connected.

- The **server module** loads, in the first place, the predictive model generated by the training module. If this model has not been generated, it will not be able to be executed. Once the model is loaded, it will remain waiting until a request is received from a client module. This request contains an image of a user face. Once the image is received, it extracts all the necessary information from the face using the same algorithm as the training module and predicts the mood of the user. This mood is the response that will be sent to the client module.

Having the training module separated from the others allows generating and saving a predictive model that later will be used by the server module, avoiding the fact of having to generate a new one each time the server starts. In this way, execution time is saved, since training a model is the task that takes the longest time and the one that requires the most computing capacity during the whole execution of the system.

On the other hand, the fact that most of the new manufactured passenger cars have internet connection, has led to the separation of the client and server modules, allowing the server module to be set in a different computer with greater computing capacity and being able to be accessed by the client modules of different vehicles. It should be borne in mind that this does not prevent the server module from being housed into the own vehicle computer so that it has the possibility of acting when there is no connection to internet or both modules cannot connect.

In addition, if updating the predictive model would be wanted, it only had to be updated the server module, either loading the new model or updating the information extraction algorithm of the faces. And, if the server were housed in an external computer and not in the own vehicle, it would not imply the need to produce an update to install on the computer of the vehicle, so that it would not affect the final user.

Generating the prediction model

The first step to generate a predictive model is finding a facial database classified by moods. The database used has been the *extended Cohn-Kanade* database. This database contains images of the faces of 122 subjects with 7 different moods of which 6 are used, since a category, specifically contempt, did not contain enough samples to train the model. Therefore, the moods that the system will predict are: anger, disgust, fear, happiness, sadness and surprise. The following illustration shows a sample of each mood used:



Illustration XXVII. Sample images of the categories used in the database. From left to right: Anger, disgust and fear in the first row; happiness, sadness and surprise in the second.

In addition, the database contains an image of each subject with a neutral gesture. Giving rise to a database of 488 images for 7 different moods.



Illustration XXVIII. Sample image of a neutral face from the database.

Once the database of face images is ready and classified by moods with enough samples, the next step is making the computer recognize the face in each image. For this task, it was decided to use some functions of the artificial vision and machine learning libraries OpenCV and dLib, since they offer great support for these tasks.

When a face is detected in an image, four points that form a rectangle are obtained, determining the position of the image where the face has been found.



Illustration XXIX. Sample image for facial detection.



Illustration 30. Sample image with detected face.

Once the position of the face inside the image has been identified, the system can already extract all the information required from it.

It should be noted that both the characteristics extracted from each image and the classification algorithm used affect directly the probability of success in the prediction of mood; so this is the most important part in the development of the system to bear in mind in order to achieve the best behavior possible of the whole system.

Is worthy of mentions, moreover, that both, the classification algorithm and the characteristics, will also affect the time required to train the model, such as the time needed to carry out the prediction.

After the realization of different tests in which different algorithms were evaluated, using different characteristics extracted from the faces for the training of the system and later to determine the facial expression, it has been decided to carry put the following solution:

- The characteristics extracted from the image are:
 - The *Facial Landmarks*, sixty-eight points that delimit the contours of the most important elements in facial gesturing: the first 17 points delimit the contour of the face; from point 18 to 22 the right eyebrow; from 23 to 27 the left eyebrow; from 28 to 36 the nose; from 37 to 42 to the right eye and from 43 to 48 to the left; and, finally, the points from 49 to 68 delimit the lips.



Illustration XXXI. Facial Landmarks example.

- The nose points are used to determine the possible inclination of the face and correct it.
- The distance of each Landmark Facial to the midpoint of the face.
- The inclination held by each Landmark Facial in relation to the midpoint of the face.
- The algorithm used to train the model is the Support Vector Machine with a linear kernel function, since comparing the results with the other algorithms used in the different tests performed, it is the one that obtains the best results, as well as being the most efficient.

Bearing all this information in mind, it resulted a percentage of success of 80% in the later predictions. But if the results are analyzed, observing the confusion matrix and carrying out a series of tests, it was determined that the angry and disgusted classes contributed confusion when determining the moods. This may be due to the similarity of the faces when both gestures are set, as well as maybe lacking of enough sample images. Looking at the images shown below, it can be concluded that it is most likely due to the similarity of both gestures:



Illustration XXXII. Sample images: Anger and Disgust.

In this way, it has been concluded that, uniting both categories into one, better results are obtained. Giving rise to a percentage of final success of 86%.

System operation

Once the predictive model has been trained and saved in a file, the operation of the system is explained below:

First, the server module must be executed. This module works like a web application. First, it will load the predictive model before being able to predict moods. The server, if executed on an external computer, will be continuously waiting to receive requests from other client modules that can connect to it; on the other hand, if it is executed locally (on the vehicle itself), the client and server will be running on the same network, so it will be waiting until the client module detects and sends the face of the driver of the vehicle.

With the server properly started and running, the client part can already be started. When it starts the first action that takes place is to turning on the camera. The camera starts taking pictures until a face is detected. Whenever the user's face is detected, the area of the image where the face is located is cut out and sent to the server.

When the server receives a request from the client with the image of a face, it extracts the same characteristics that were extracted from each face previously when the predictive model was trained. That is, the Facial Landmarks are extracted, the inclination of the face is corrected and the distance of each Landmark Facial is saved as well as its inclination in relation to the midpoint of the face.

Once all the described features have been extracted, the server module already has the necessary information so that the predictive model can determine the mood of the received image.

Finally, the prediction result is sent to the client, so that the vehicle may take some action; and the server processes the execution data in a log.

Below is a diagram that shows the described system operation:

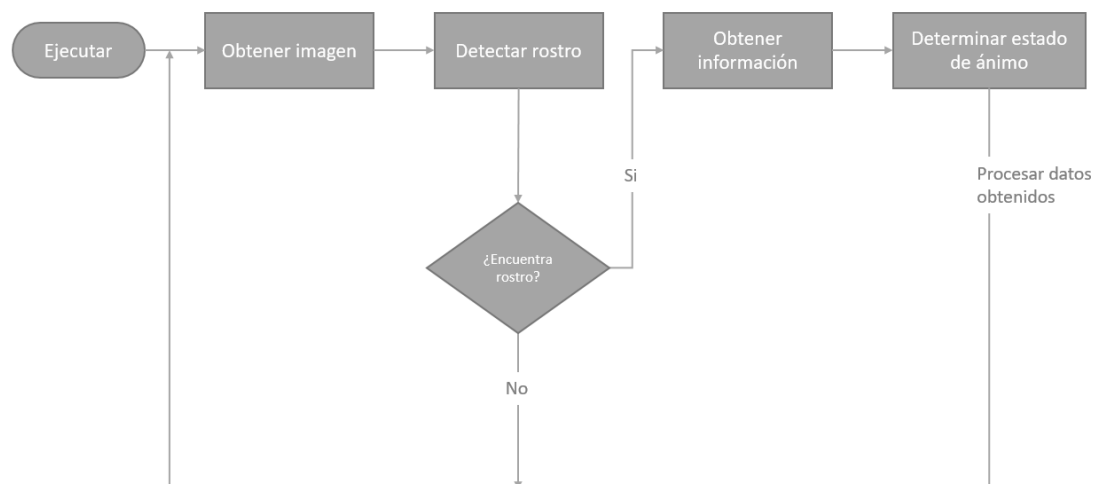


Illustration XXXIII. System execution diagram.

Client-Server communication

As mentioned above, the server module was created as a web application. In this way, if the server were in an external device, multiple client modules could access from different devices to consume its resources if their applications had an Internet connection. If connectivity is not available, a server module housed in the vehicle itself would be used, so that both modules

would be in the same network. In both cases, the exchange of information is carried out in the same way:

The client module must identify a face in the image. When it identifies one, it cuts it out and prepares it to send it to the server. The image, to be able to be transmitted, is introduced in a json, which is a light text format created for the exchange of data between web applications. In this way, the image is already available in a correct format to be transmitted to the server. In order for the server to predict the mood of that image, the client must send a Post request of "predict" with the previous information.

On the other hand, when the server module has been started and is running correctly, it is waiting for requests from a client module, either running locally or online. As mentioned above, the client modules will always receive "predict" requests with a json in which the image of a face that has already been cut when it was found. As the information is contained in a json, the first thing to do in order to extract information from the image is to convert it to a correct format, that is, to a matrix. After this, information can already be extracted and the mood of the face determined as explained previously. Finally, the resulting mood is sent to the client module in a json object again.

Conclusions

The work has described the development of an advanced driving assistance system (ADAS) capable of recognizing the mood of a vehicle driver. At this time, it is necessary to check if the expected objectives of the project have been completed and to determine whether, in effect, the system is capable of recognizing the mood of a driver. To do this, it is necessary to start from the most basic tasks performed until, in its last stage, check whether the final purpose has been fulfilled.

First of all, it has been managed to train a system capable of determining the mood of a person through his face with almost 500 images for a total of 6 emotions. To do this, different tests have been carried out and both the algorithm that gave the most success and the most relevant information to determine the facial expression through images of a person's face have been obtained. This has been the hardest part to perform, besides being the one that has carried out the most research work; since generating a good predictive model was fundamental for all subsequent work. Also, note that finding a database of suitable images for this has been costly.

From here, generating the rest of the system has required work, but the basis of the project was already solid. Then connect a part that controls a camera and detects faces with another able to obtain information from those faces and determine their mood were the next steps required. For this, Flask, from Python, was the framework used, allowing the different parts of the system to be executed in different devices.

Executing the different parts of the system in different devices was an important task in order to achieve an efficient operation of the system, without overloading the computer of the vehicle itself. This task has been carried out with a correct flow of images. And, to highlight, the result is equally satisfactory when the different parts of the system are executed in the same device, although said flow will surely be somewhat reduced when it is executed in devices with fewer resources than those used for the developing of the project.

Finally, at the technical side, highlight that the main objective of the system has been met and the mood of a driver is detected quite accurately, allowing the vehicle to be the one that control this information now. This way, in the future, the vehicle may take actions from this.

In the personal side, it is remarkable to mention the awareness of the difficulty and the work involved in carrying out a project of this type. In spite of having had previous contact with the development of applications in which the use of artificial vision and machine learning was necessary, the fact that on this occasion it was about monitoring the human face and, from this, obtaining its emotion has had a point of extra difficulty. Despite this, the result has been satisfactory and the realization of the project has been a very interesting challenge.

Finally, mention that the different knowledge acquired during the degree in computer engineering has been useful for the development of the project and that, without the time of internships in the company and the knowledge acquired in the department of Everis Spain for the development of artificial intelligence projects, the completion of the project would have been significantly harder.

Future work

The advanced system of assistance to driving that has been developed is a first and small approach to the monitoring of the user of a vehicle. There are different aspects in which future work could deepen more and keep developing:

- First of all, the driver of the vehicle could be identified so as his distraction level could also be monitored. Making the system able to determine if the user is asleep or if he is focused on driving.
- The following step could be also monitoring the behavior and the mood of the other occupants of the vehicle.
- Another element to bear in mind in the development would be to make the system capable of determining the mood of the users also in adverse lighting situations, such as at night or in a tunnel.
- Getting a database with a greater number of face samples for the different emotions detected could also provide an improvement in the accuracy when detecting moods.
- Making the most complex calculations carried out with the images to be fulfilled by the graphic card (GPU) instead of the CPU would greatly optimize the performance of the system in general.
- Finally, if the system could have control over different elements of the vehicle, whether the radio, the position of the seats, mirrors, even at some point the brake could be a very attractive aspect in vehicles that had the system integrated.

Keywords: Computer vision, Machine learning, Face recognition, Advanced driver assistance systems.

Resumen

Introducción

Los accidentes de tráfico siguen siendo una de las principales causas de mortalidad en la actualidad. En el mundo fallecen alrededor de 1.250.000 personas a causa de estos accidentes, pero cabe destacar que el índice de mortalidad por estos accidentes varía en gran medida en función de la región geográfica.



Ilustración XXXIV. Fallecidos por accidente de tráfico en función de la región (Fuente: www.revista.dgt.es).

Como se puede observar en la *Ilustración I*, la probabilidad de que un accidente de tráfico sea mortal es mucho mayor en regiones como África o el Mediterráneo Oriental que en Europa, por ejemplo. Esta diferencia se debe, en gran medida, a la antigüedad media de los automóviles de cada región, a los sistemas de seguridad que incluyen estos automóviles y a la falta de regulación de muchos países. Europa, en comparación con la mayoría de las demás regiones geográficas mostradas en el mapa, es un área de mayor riqueza y con un mayor grado de desarrollo. El resto de regiones, contemplan gran cantidad de países subdesarrollados o que aún están en vías de desarrollo, pudiendo concentrarse la riqueza en ciertos puntos, pero no siendo algo corriente. Esto permite que, en áreas más ricas la antigüedad media de los automóviles como en Europa, sea mucho menor que en África. Además, teniendo esto en cuenta, la incorporación generalizada de elementos de seguridad básicos, como puede ser el cinturón de seguridad, no

se hizo obligatorio en España hasta 1990; esto significa que la mayoría de vehículos de fabricación previa ni siquiera disponían de este elemento de seguridad.

En gran parte de estas regiones, el uso de elementos de seguridad básicos hoy en nuestra cultura, aun no es obligatorio. Y dada la antigüedad de sus vehículos, muchos carecen también de mecanismos de seguridad básicos.

Por otro lado, el 49% de las víctimas mortales por accidentes de tráfico son peatones, ciclistas y motociclistas, lo cuales no cuentan con mecanismos de seguridad tan avanzados como de los que puede disponer un coche. Además, cabe mencionar también que los peatones corresponden con el 22% de todas las víctimas que tienen lugar en estos accidentes, sin ocupar si quiera el asiento de un vehículo.

Cabe destacar que el factor humano sigue siendo el origen del 90 % de los accidentes de tráfico: ya sea por distracción, factor emocional, conducción bajo los efectos de alcohol o drogas, exceso de velocidad, etc. Esto sugiere que gran parte de estos accidentes podrían evitarse con el uso sistemas de seguridad que ayudasen al conductor a mantener la atención en la conducción, para reducir los accidentes que se puedan producir por distracción del conductor; y con una regulación más rigurosa para los accidentes que potencialmente pudieran producirse por el incumplimiento de las leyes.

De este modo, investigando y desarrollando nuevos sistemas que aporten seguridad a los vehículos de fabricación actual, que además ayuden al conductor de un vehículo a reducir las situaciones de distracción y a aumentar su atención en la conducción, no solo se reduciría la cantidad de accidentes, sino que también la cantidad de fallecimientos por estos. Para esto es importante también que la regulación del uso de elementos de seguridad en vehículos mejore y que su incorporación sea económicamente sostenible y accesible para todos.

Objetivos

El objetivo principal de este proyecto consiste en el desarrollo de un Sistema Avanzado para la Asistencia a la Conducción, que, prediciendo el estado emocional del conductor, ayude a reducir las situaciones de distracción al volante de un vehículo.

Este sistema monitorizaría a través de una cámara rostro del conductor, pudiendo determinar el estado de ánimo de éste. De esta forma, el vehículo tendrá información sobre el estado emocional del conductor y podría determinar la predisposición a distraerse de este; pudiendo llegar a tomar acciones en el interior del vehículo para reducir esta mala predisposición.

Por otro lado, destacar que el sistema, además de permitir generar una conducción más serena y con menos distracciones, pueda dar lugar a una mejora en la experiencia de la conducción del vehículo, ya que podría dar lugar a un alto grado de personalización y adaptación para cada usuario.

Arquitectura

El sistema se compone de tres módulos. Cada uno de ellos realiza principalmente una función y la comunicación entre ellos se realiza siempre a partir del *módulo servidor*, que será uno de estos módulos. Los otros dos módulos serán el *módulo de entrenamiento* y el *módulo cliente*. En el siguiente diagrama se muestran los módulos y sus relaciones:

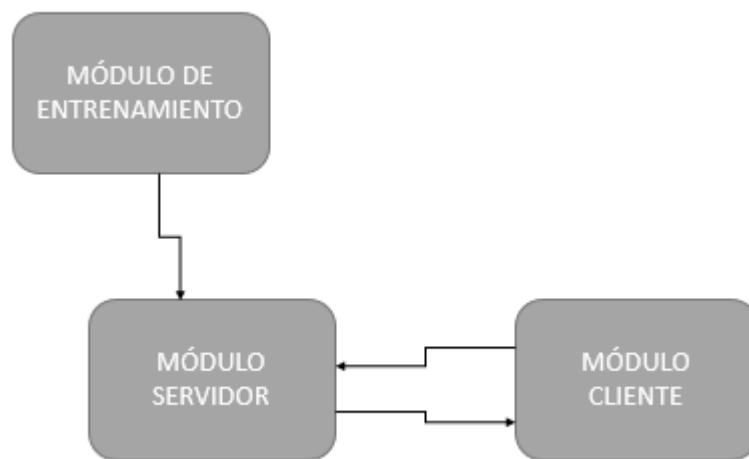


Ilustración XXXV. Diagrama de Arquitectura del sistema.

- El **módulo de entrenamiento** contiene el algoritmo que se usará para extraer información de las imágenes de los rostros de los usuarios. Requiere la existencia de una base de datos de rostros clasificada por estados de ánimo. A partir del algoritmo y la base de datos, el módulo de entrenamiento es capaz de generar un modelo predictivo que posteriormente será usado por el módulo servidor para determinar el estado de ánimo de los usuarios.
- El **módulo cliente** se encargará principalmente del manejo de la cámara. Inicia la cámara y extrae un flujo continuo de imágenes de ella. Si detecta un rostro en la imagen, recorta el área de este, lo envía al módulo servidor y espera la respuesta de este. La respuesta

contendrá el estado del ánimo del rostro que fue detectado. Este módulo, además, controla los posibles errores de ejecución que puedan tener que ver con la cámara producidos tanto en tiempo de ejecución, como en caso de no estar conectada.

- El **módulo servidor** carga, en primer lugar, el modelo predictivo generado por el módulo de entrenamiento. Si no se ha generado dicho modelo, no podrá ejecutarse. Una vez cargado el modelo, espera a recibir una imagen de un rostro del módulo cliente. Cuando recibe esta imagen extrae toda la información necesaria del rostro empleando el mismo algoritmo que el módulo de entrenamiento y predice el estado de ánimo del usuario. Este estado de ánimo es la respuesta que se le envía al módulo cliente.

Disponer del módulo de entrenamiento separado de los demás permite generar y guardar un modelo predictivo que posteriormente será usado por el módulo servidor, evitando que se tenga que generar de nuevo que el servidor se inicia. De esta forma se ahorra tiempo de ejecución, pues es la tarea que más tiempo lleva realizar y la que más capacidad de cómputo necesita.

Por otro lado, el hecho de que gran parte de los turismos de nueva fabricación dispongan de conexión a internet ha motivado a separar los módulos cliente y servidor, permitiendo alojar el módulo servidor en un computador con mayor capacidad de cálculo y que este sea accedido desde los módulos clientes de distintos clientes. Debe tenerse en cuenta que esto no impide que se aloje un módulo servidor en el computador del propio vehículo para que actúe cuando no se dispone de conexión a internet.

Además, si se deseara actualizar el modelo predictivo, solo habría que actualizar el servidor, ya fuera cargando el nuevo modelo o actualizando el algoritmo de extracción de información de los rostros. Y, si el servidor se encontrase en un computador externo y no en el del propio vehículo, no implicaría la necesidad de producir una actualización que instalar en el computador del propio vehículo, de forma que no afectaría al usuario final.

Generación del modelo predictivo

El primer paso para generar un modelo predictivo es disponer de una base de datos con rostros clasificada por estados de ánimo. La base de datos empleada ha sido la base de datos extendida de *Cohn-Kanade*. Esta base de datos contiene imágenes de rostros de 122 sujetos con 7 estados

de ánimo diferentes de los cuales se usan 6, ya que una categoría, concretamente deprecio, no contenía muestras suficientes para entrenar el modelo. Por tanto, los estados de ánimo que el sistema va a predecir son: enfado, disgusto, susto, alegría, tristeza y sorpresa. En la ilustración que aparece a continuación se expone una muestra de cada estado de ánimo usado:



Ilustración XXXVI. Imágenes de muestra de las categorías empleadas de la base de datos. De izquierda a derecha: Enfado, disgusto y susto en la primera fila; alegría, tristeza y sorpresa en la segunda.

Además, contiene una imagen de cada sujeto con un gesto neutral. Dando lugar a una base de datos de 488 imágenes para 7 estados de ánimos diferentes.



Ilustración XXXVII. Imagen de muestra de la base de datos de un rostro neutral.

Una vez se dispone de una base de datos de imágenes de rostros clasificada por estados de ánimo con muestras suficientes, el siguiente paso es reconocer el rostro en cada imagen. Para esta tarea se emplean funciones propias de las librerías de visión artificial y aprendizaje automático *OpenCV* y *dLib*, ya que ofrecen gran soporte para estas tareas.

Cuando se detecta un rostro en una imagen, se obtienen cuatro puntos que forman un rectángulo determinando la posición en la que el rostro ha sido encontrado.



Ilustración XXXVIII. Imagen de muestra para detectar un rostro.



Ilustración XXXIX. Imagen de muestra con rostro detectado.

Una vez se ha identificado la posición del rostro dentro de la imagen, ya se puede extraer información de este.

Cabe destacar que tanto las características que se extraen de cada imagen como el algoritmo empleado de clasificación afectan directamente a la probabilidad de acierto en la predicción del estado de ánimo; de modo que esta es la parte más importante en el desarrollo para lograr el mejor comportamiento del sistema posible.

Mencionar, además, que tanto el algoritmo de clasificación como las características afectarán también al tiempo requerido para entrenar el modelo, como el tiempo necesario para realizar la predicción.

Tras la realización de distintas pruebas en las que se evalúan diferentes algoritmos, utilizando características diferentes de los rostros para el entrenamiento del sistema y posteriormente determinar el estado de ánimo, se ha decidido tomar la siguiente solución:

- Las características que se extraen de la imagen son:
 - Los *Facial Landmarks*, sesenta y ocho puntos que delimitan los contornos de los elementos más importantes en la gesticulación facial: los primeros 17 puntos

delimitan el contorno de la cara; del punto 18 al 22 la ceja derecha; del 23 al 27 la ceja izquierda; del 28 al 36 la nariz; del 37 al 42 al ojo derecho y del 43 al 48 el izquierdo; y, por último, los puntos del 49 al 68 delimitan lo labios.

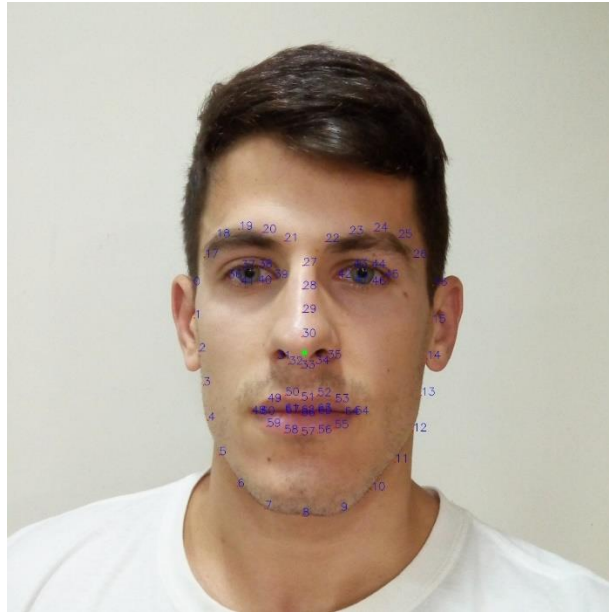


Ilustración XL. Ejemplo de Facial Landmarks

- Se usan los puntos de la nariz para determinar la posible inclinación de la cara y corregirla
- Distancia de cada *Facial Landmark* al punto medio del rostro.
- Inclinación que guarda cada *Facial Landmark* respecto al punto medio de la cara.
- El algoritmo empleado para entrenar el modelo es el Support Vector Machine con una función de kernel lineal, pues comparando los resultados con los demás algoritmos en las pruebas realizadas, es el que mejor resultados obtiene, además de ser el más eficiente.

Con esto se obtuvo un porcentaje de acierto de un 80% en las predicciones. Pero si se analizan los resultados, observando la matriz de confusión y realizando una serie de pruebas, se determinó que los estados enfado y disgusto aportaban confusión a la hora de determinar los estados de ánimo. Esto puede deberse tanto por similitud de los rostros cuando se realizan

ambos gestos, como por disponer de imágenes de muestra insuficientes. Si se observa las imágenes que se muestran a continuación, se puede concluir que muy probablemente sea por la similitud entre ambos gestos:



Ilustración XLI. Imágenes de muestra: Enfadado y Disgustado.

De este modo, se ha concluido que, uniendo ambas categorías en una sola, se obtienen mejores resultados. Dando lugar a un porcentaje de acierto final de 86%.

Funcionamiento del sistema

Una vez entrenado el modelo predictivo y guardado en un archivo, el funcionamiento del sistema es el siguiente:

En primer lugar, debe ejecutarse el módulo servidor. Este módulo funciona como una aplicación web. Primero cargará el modelo predictivo para después poder predecir estados de ánimo. El servidor, si se ejecuta en un computador externo, estará esperando continuamente a recibir peticiones de otros módulos clientes que puedan conectarse a él; por otro lado, si se ejecuta en local (en el propio vehículo), cliente y servidor se encontrarán ejecutando en la misma red y estará esperando hasta que el módulo cliente detecte el rostro del usuario conductor del vehículo.

Con el servidor correctamente iniciado y funcionando, la parte cliente ya puede iniciarse. Cuando ésta se inicia la primera acción que lleva a cabo es encender la cámara. La cámara comienza a tomar imágenes hasta que se detecta un rostro. Cuando se detecta el rostro del usuario, se recorta el área de la imagen en la que se encuentra el rostro y se le envía al servidor.

Cuando el servidor recibe una petición del cliente con la imagen de un rostro, extrae las mismas características que se extrajeron de cada rostro anteriormente para entrenar el modelo predictivo. Es decir, se extraen los *Facial Landmarks*, se corrige la inclinación de la cara y se guarda tanto la distancia de cada *Facial Landmark* como su inclinación respecto al punto medio de la cara.

Una vez extraídas todas las características descritas, el módulo servidor ya dispone de la información necesaria para que al modelo predictivo pueda determinar el estado de ánimo del rostro de la imagen recibida.

Por último, se le envía el resultado de la predicción a la parte cliente, de forma que pueda tomar alguna acción; y el servidor procesa los datos de ejecución en un log.

A continuación, se adjunta un diagrama que muestra el funcionamiento descrito del sistema:

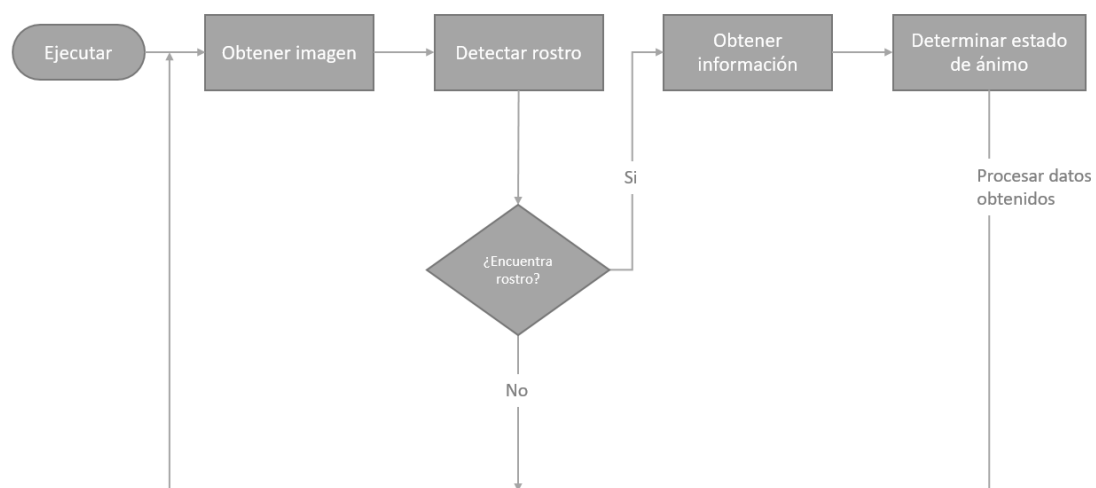


Ilustración XLII. Diagrama de ejecución del sistema.

Comunicación cliente-servidor

Como se ha mencionado anteriormente, el módulo servidor se ha creado como una aplicación web. De esta forma, si el servidor se encontrase en un dispositivo externo, múltiples módulos clientes podrían acceder desde distintos dispositivos para consumir sus recursos si sus aplicaciones dispusieran de conexión a internet. De no disponer de conectividad, se usaría un

módulo servidor alojado en el propio vehículo, de forma que se encontrarían en la misma red. En ambos casos, el intercambio de información se lleva a cabo de la misma forma:

El módulo cliente debe identificar un rostro en la imagen. Cuando identifica uno, este lo recorta y lo prepara para enviárselo al servidor. La imagen, para poder ser transmitida, se introduce en un *json*, que es un formato de texto ligero creado para el intercambio de datos entre aplicaciones web. De este modo, ya se dispone de la imagen en un formato correcto para ser transmitida al servidor. Para que el servidor prediga el estado de ánimo de esa imagen, el cliente debe enviarle una petición "*predict*" de tipo *Post* con la información anterior.

Por otro lado, cuando el módulo servidor se ha iniciado y está ejecutándose de forma correcta, está esperando peticiones de algún módulo cliente, ya sea ejecutando en local o en línea. Como se ha mencionado anteriormente, de los módulos clientes va a recibir siempre peticiones "*predict*" con un *json* en el que se encuentra la imagen de un rostro ya recortado. Como la información va contenida en un *json*, lo primero que debe hacer para poder extraer información de la imagen es convertirla a un formato correcto, es decir, a una matriz. Tras esto, se puede extraer la información y determinar el estado de ánimo del rostro como se ha explicado anteriormente. Por último, envía al cliente el estado de ánimo resultante en un *json* de nuevo.

Conclusiones

En el trabajo se ha descrito el desarrollo de un sistema avanzado de asistencia a la conducción capaz de reconocer el estado de ánimo de un conductor. En este momento, es necesario comprobar si se han completado los objetivos esperados del proyecto y determinar si, en efecto, el sistema es capaz de reconocer el estado de ánimo de un conductor. Para ello, se debe comenzar desde las tareas realizadas más básicas hasta, en su última etapa, revisar si el objetivo final se ha cumplido.

Para comenzar, se ha conseguido entrenar un sistema capaz de determinar el estado ánimo de una persona a través su rostro con más de 500 imágenes para un total de 6 emociones. Para ello se han realizado distintas pruebas y se ha obtenido tanto el algoritmo que mayor acierto daba como la información más relevante para determinar el estado de ánimo a través de imágenes del rostro de una persona. Esta ha sido la parte más costosa de realizar, además de ser la que más trabajo de investigación ha conllevado; pues generar un buen modelo predictivo era fundamental para todo el trabajo posterior. Además, destacar que encontrar una base de datos de imágenes adecuadas para esto ha sido costoso.

A partir de aquí, generar el resto del sistema ha requerido trabajo, pero la base del proyecto ya era sólida. Entonces conectar una parte que controla una cámara y detecta rostros con otra capaz de obtener información de dicho rostro y determinar su estado de ánimo eran los siguientes pasos requeridos. Para ello se empleó *Flask*, de *Python*, permitiendo ejecutar las distintas partes del sistema en dispositivos diferentes.

Ejecutar las distintas partes del sistema en dispositivos diferentes era una tarea importante para lograr un funcionamiento eficiente del sistema, sin sobrecargar el del propio vehículo. Esta tarea se ha logrado llevar a cabo con un flujo de imágenes correcto. Y, destacar, que el resultado es igualmente satisfactorio cuando las distintas partes del sistema se ejecutan en el mismo dispositivo, aunque dicho flujo seguramente se vea algo reducido cuando se ejecute en dispositivos con menos recursos que los empleados para la realización del proyecto.

Para finalizar a nivel técnico, destacar que se ha cumplido el objetivo principal del sistema y se detecta de forma bastante precisa el estado de ánimo de un conductor, permitiendo ahora al vehículo controlar esta información. De esta forma, en un futuro, el vehículo podrá tomar acciones a partir de esto.

En el aspecto personal, se ha tomado conciencia de la dificultad y del trabajo que conlleva realizar un proyecto de este tipo. A pesar de haber tenido contacto anterior con el desarrollo de aplicaciones de en el que el uso de *visión artificial* y *machine learning* fuera necesario, el hecho de que en esta ocasión se tratara de monitorizar el rostro humano y, a partir de este, obtener su emoción ha tenido un punto de dificultad añadida. A pesar de esto, el resultado ha sido satisfactorio y la realización del proyecto ha sido un reto muy interesante.

Por último, mencionar que los distintos conocimientos adquiridos durante el grado en ingeniería informática han sido útiles para el desarrollo del proyecto y que, sin el tiempo de prácticas en empresa y el conocimiento adquirido en el departamento de *Everis Spain* destinado al desarrollo de proyectos de inteligencia artificial, la realización del proyecto hubiera sido notablemente más costosa.

Trabajos futuros

El sistema avanzado de asistencia a la conducción que se ha desarrollado es una primera y pequeña aproximación a la monitorización del usuario de un vehículo. Hay distintos aspectos en los que se podría profundizar más y seguir desarrollando:

- En primer lugar, se podría identificar al conductor del vehículo y monitorizar también su nivel de distracción. Que el sistema sea capaz de determinar si el usuario está dormido o si tiene su atención puesta en la conducción.
- Lo siguiente podría ser monitorizar tanto el comportamiento como el estado de ánimo de los demás ocupantes del vehículo.
- Otro elemento a tener en cuenta en el desarrollo sería conseguir que el sistema fuera capaz de determinar el estado de ánimo de los usuarios también en situaciones lumínicas adversas, como por ejemplo durante la noche o en un túnel.
- Obtener una base de datos con mayor cantidad de muestras de rostros para las distintas emociones detectadas podría aportar también una mejora en la precisión a la hora de detectar estados de ánimo.
- Conseguir que los cálculos más complejos que se realizan con las imágenes fueran llevados a cabo por la tarjeta gráfica en lugar del procesador optimizaría en gran medida el rendimiento del sistema en general.
- Por último, que el sistema pudiera llegar a tener control sobre distintos elementos del vehículo, ya fuera la radio, la posición de los asientos, retrovisores, incluso en algún momento el freno podría ser un aspecto muy atractivo en vehículos en los que ya se hubiera integrado el sistema.